AD-A206 326

(2)

TECHNICAL REPORT CR-RD-GC-88-1

SYMBOLIC ANALYSIS OF MULTIVARIABLE MISSILE SYSTEMS

Chih-Fan Chen
Metropolitan College
Boston University
Boston, MA 02215

and

Ming M. Chen
College of Engineering
Boston University
Boston, MA 02215

MAY 1988

Prepared for:
Guidance and Control Directorate
Research, Development, and
Engineering Center

Contract No. DAAH01-84-C-0489

U.S. ARMY MISSILE COMMAND
Redstone Arsenal, Alabama 35898-5000

Cleared for public release; distribution is unlimited.

DISPOSITION INSTRUCTIONS

DESTROY THIS REPORT WHEN IT IS NO LONGER NEEDED. DO NOT RETURN IT TO THE ORIGINATOR.

DISCLAIMER

THE FINDINGS IN THIS REPORT ARE NOT TO BE CONSTRUED AS AN OFFICIAL DEPARTMENT OF THE ARMY POSITION UNLESS SO DESIGNATED BY OTHER AUTHORIZED DOCUMENTS.

TRADE NAMES

USE OF TRADE NAMES OR MANUFACTURERS IN THIS REPORT DOES NOT CONSTITUTE AN OFFICIAL INDORSEMENT OR APPROVAL OF THE USE OF SUCH COMMERCIAL HARDWARE OR SOFTWARE.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188
Exp. Date Jun 30, 1986

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Cleared for public release; distribution is unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S) CR-RD-GC-88-1
6a. NAME OF PERFORMING ORGANIZATION Boston University, Dept of Applied Science 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION US Army Missile Command
6c. ADDRESS (City, State, and ZIP Code) Boston, MA 02215	7b. ADDRESS (City, State, and ZIP Code) Redstone Arsenal, AL 35898
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER DAAH01-84-C-0489
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. 62303A; PROJECT NO. 3A214; TASK NO. 00; WORK UNIT ACCESSION NO DA304514

11. TITLE (Include Security Classification)

SYMBOLIC ANALYSIS OF MULTIVARIABLE MISSILE SYSTEMS

12. PERSONAL AUTHOR(S)

Chih-Fan Chen, and Ming M. Chen

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM 84/7/1 TO 85/12/3	1988/May/04	167

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Non-linear systems, Multi-linear systems, Control systems, Grassman algebra, Dead Time Compensators, Smith Compensators

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

A new systematic procedure mainly based on the Grassmann algebra is established for evaluating a symbolic transfer function in particular and a transfer matrix in general. After illustrating the rules for symbolic determinant evaluation, we extend the method to multivariable continuous and hybrid systems by the decoupling techniques. P-constrained and V-constrained models have been studied on the one hand; dead time compensators and the Smith compensators are deeply investigated on the other as the generalization of the new computerized symbolic treatment technique. In the appendix, this Grassman algebra-based method is compared with the existing methods such as symbolic evaluation via Fourier transform, symbolic evaluation via number theoretic transform, etc.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL G. Bradford	22b. TELEPHONE (Include Area Code) 205-876-7081 22c. OFFICE SYMBOL AMSMI-RD-GC-S

DD FORM 1473, 84 MAR

83 APR edition may be used until exhausted.
All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

SYMBOLIC ANALYSIS OF
MULTIVARIABLE MISSILE SYSTEMS

Submitted to

U.S. Army Missile Command
Huntsville, Alabama

by

Professor C. F. Chen
Department of Computer Science
Metropolitan College
Boston University
Boston, MA 02215

Professor Ming M. Chen
Department of Aerospace and
Mechanical Engineering
College of Engineering
Boston University
Boston, MA 02215

MAY 1988

TABLE OF CONTENT

1.0 Introduction

A physical system in general consists of hundreds of parts which are connected in various forms. The governing equations for the system can be formulated in a number of distinct ways. For example, Lagrange's approach[1], which is based on energy, and the topology approach[2], which is based on casual patterns, are well known. The governing equations obtained are nonlinear in general and the size is usually very large.

Let us look into the topological approach. The development of governing equations depends on:

(a) the structure of the system and
(b) the properties of the parts.

The laws established for dealing with structures are Kirchoff's law, D'Alambert's principle, etc., while the laws for treating devices are Ohm's law, and Hooke's law, etc. We have the former laws of constraints, and the latter laws of performance.

Nonlinear phenomena such as saturation, dead-time and backlash[3] frequently appear in the laws of performance and nonlinear junction structure or geometric nonlinearity quite often occur in the laws of constraint[4].

After the governing equations has been developed, the first treatment to these equations is usually to substitute numerical values and to linearalize them. The solutions of course, are easy to obtain, but the results are far from the reality and sometimes lead to wrong conclusions, this is particularly true for most practical large scale systems[5].

In order to keep the original modelling, the engineer should keep symbolic analysis as long as he/she can. In other words, to substitute numerical values and to perform linearizations as late as possible.

However, systems to be investigated become larger and larger and computer's capability for treating symbols has not been improved. Therefore, there is a gap between the system modelling and qualitative analysis. there is considerable and growing interest[6] in establishing symbolic analysis especially by computer-aided procedures in order to close this gap. Recent examples are computerization of Mason's signal flow graph and Coates flow formula[7], development of tree enumeration methods[8], and establishment of parameter extraction techniques[9]. However, these are all for linear models and their procedures are difficult to extend to the nonlinear domain. It seems that there is a missing link between symbolic analyses of linear models and nonlinear models. What we need is to develop a general scheme to cope with the fundamental problem---symbolic analysis of large scale linear and nonlinear systems.

1.1 Whitehead's Universal Algebra

Since Mason's time[7], the various methods developed for symbolic analysis are special techniques for particular types of questions or deliberately coping with certain computing machines. No general foundation has ever been established. This is why the missing link from the nonlinear models to linear ones has never been found.

Before writing the well-known classical work, Principia of Mathematica with Russell, Whitehead wrote a book, 'A Treatise on Universal Algebra'[10] which presents a thorough investigation of the various systems of symbolic reasoning allied with ordinary algebra. Whitehead treated three basic schools:

(1) Boolean algebra,

(2) Hamilton's formulation and

(3) Grassmann algebra.

His insight was far ahead of his time. In these recent years, Boolean algebra in digital computers and Hamilton's formulation in control and optimization theory have played important roles. To the authors of this report, it seems that the third topic of Whitehead---Grassmann algebra is a powerful tool for investigating systems but has not been further developed since his time.

In a preliminary study, the authors found that based on the new algebra we can cope with the symbolic investigation of large scale linear systems and with further and deeper studies, we may approach some nonlinear cases.

1.2 Grassmann Algebra for Determinants

Let us briefly introduce some of Grassmann's[10] basic rules. Suppose we want to evaluate the determinant of a matrix M, where

$$M = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots\cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots\cdots & a_{2n} \\ . & . & . & & . \\ . & . & . & & . \\ . & . & . & & . \\ a_{n1} & a_{n2} & a_{n3} & \cdots\cdots & a_{nn} \end{bmatrix} \quad (1)$$

First of all, each column is expressed, in vector notation, a_i, or

$$M = [\ a_1,\ a_2,\ a_3,\ \ldots\ ,\ a_n\ ] \tag{1a}$$

where a_i is the column vector of the i^{th} column. The determinant of M, or Δ=det M, is equal to the exterior product of the vectors:

$$\Delta = a_1 \Lambda a_2 \Lambda \ldots \Lambda a_n \tag{2}$$

where Λ is a symbol denoting exterior of outer product subject to the following rules:

$$a_i\ \Lambda\ a_i = 0 \tag{3a}$$

$$(a_i\ \Lambda\ a_j) + (a_j\ \Lambda\ a_i) = 0 \tag{3b}$$

$$(a_i\ \Lambda\ a_j)\ \Lambda\ a_k = a_i\ \Lambda\ (a_j\ \Lambda\ a_k) \tag{3c}$$

$$a_i\ \Lambda\ (a_j + a_k) = (a_i\ \Lambda\ a_j) + (a_i\ \Lambda\ a_k) \tag{3d}$$

where a_i, a_j, and a_k are vectors defined in (1a).

The last three rules, (3b) to (3d), are the commutative property, associative property and the distributive property, respectively, while the first rule, (3a) is unique.

Let us use an example to illustrate how to apply the Grassmann rules to find the value of the following determinant.

$$M_1 = \begin{bmatrix} a_{11} & a_{12} & 0 \\ 0 & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{4}$$

We express i^{th} column by vector a_i, or

$$a_1 = (a_{11}, a_{31})$$
$$a_2 = (a_{12}, a_{22}, a_{32}) \tag{5}$$
$$a_3 = (a_{23}, a_{33})$$

It is noted that the second subscript of each term in (5) is omitted. Then the determinant of M_1 is evaluated by the outer product of the column vectors.

$$\begin{aligned}\Delta_1 = \det M_1 &= a_1 \wedge a_2 \wedge a_3 \\ &= (1,3) \wedge (1, 2, 3) \wedge (2, 3) \\ &= (11, 12, 13, 31, 32, 33) \wedge (2, 3) \\ &= (12, 13, 31, 32) \wedge (2, 3) \\ &= (122, 132, 312, 322, 123, 133, 313, 323) \\ &= (132, 312, 123).\end{aligned} \tag{6}$$

It is to be noted that when we performed the exterior product operations, we have used rule (3a). Therefore, 11, 33, 122, 322, 133, 313, and 323 become zero in various steps. Then we decode equation (6) and write it as:

$$\Delta_1 = (a_{11}\, a_{32}\, a_{23},\ a_{31}\, a_{12}\, a_{23},\ a_{11}\, a_{22}\, a_{33}) \tag{7}$$

the second subscript of each element in (7) is added according to the positions, while the first subscript in (7) is copied from (6).

For determining the sign of each item in (7), we simply check whether it is odd or even permutation. For example, in order to change 132 into the natural order 123, we need one interchange of the positions of digits. Therefore, the term 132 should be with a negative sign. Similarly, 312 should have a plus

sign, because, for changing it into a natural order, we have to interchange the positions of digits twice which means even permutation and a positive sign should be associated with the item. Finally, we have

$$\Delta_1 = -a_{11}\, a_{32}\, a_{23} + a_{31}\, a_{12}\, a_{23} + a_{11}\, a_{22}\, a_{33} \tag{8}$$

1.3 State Equation Evaluation

Consider a typical state space model

$$\dot{x} = Ax + bu \tag{9}$$

and $$y = c^T x \tag{9a}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -6 & -11 & -6 \end{bmatrix}$$

$$b = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \qquad \text{and} \qquad c = \begin{bmatrix} 5 \\ 7 \\ 0 \end{bmatrix}$$

We are particularly interested in the evaluation of the determinant

$$\Delta = (\, sI - A \,) \tag{10}$$

$$= \begin{bmatrix} s & -1 & 0 \\ 0 & s & -1 \\ 6 & 11 & s+6 \end{bmatrix}$$

This is only a particular case of (4), if we interpret the i^{th} column as a_i. Therefore,

$$\Delta = a_1 \wedge a_2 \wedge a_3 \tag{4}$$

$$= -a_{11}\, a_{32}\, a_{23} + a_{31}\, a_{12}\, a_{23} + a_{11}\, a_{22}\, a_{33}$$

$$= -(s)\ (11)\ (-1) + (6)\ (-1)\ (-1) + (s)\ (s)\ (s+6) \tag{4a}$$

$$= 11s + 6 + s^3 + 6s^2$$

$$= s^3 + 6s^2 + 11s + 6 \tag{11}$$

which is what we expected.

In this exercise, the Grassmann algebra enables us to manipulate a practical symbolic determinant easily. However, when the system is large and when we also want to preserve the original symbols, the problem is not a simple one.

The evaluation of a symbolic determinant can be extended to the transfer function evaluation. Consider the following identities

$$\det \begin{bmatrix} P & Q \\ R & T \end{bmatrix} = \det P \det \left(T - RP^{-1}Q \right) \tag{12}$$

$$\det \begin{bmatrix} P & Q \\ R & T \end{bmatrix} = \det T \det \left(P - QT^{-1}R \right) \tag{13}$$

These two identities have been derived[11].

Now we consider the set of state equations and the output equation in (9). Our problem is to obtain the corresponding transfer function Y(s)/U(s). What we need is to interpret the two identities (12) and (13) properly and apply grassmann's algebra.

Let $R=-c^T$, $P=(sI-A)$, $Q=b$ and $T=1$. Substituting the new notation into (12) and (13) we obtain,

$$1 + c^T(sI-A)^{-1} = \frac{\det\{(sI-A)+bc^T\}}{\det(sI-A)} \qquad (14)$$

This equation was derived in a different way by Patel and was established as above and used in the deadbeat systems designed by Chen, Tsay and Yates[11].

The left hand side of (14) is the required transfer function plus one, while the right hand of (14) is simply a ratio of two determinants. Therefore, to evaluate a transfer function from a state space model requires the evaluation of two symbolic determinants.

For this problem

$$\det(sI-A) = \Delta_d = \det\begin{bmatrix} s & -1 & 0 \\ 0 & s & -1 \\ 6 & 11 & s+6 \end{bmatrix} \qquad (15)$$

and

$$\det\{(sI-A)+bc^T\} = \det\begin{bmatrix} s & -1 & 0 \\ 0 & s & -1 \\ 6+5 & 11+7 & s+6 \end{bmatrix} \qquad (16)$$

The two determinants can be solved by the Grassmann algebra method. Therefore

$$\det (sI-A) = -a_{11}\, a_{32}\, a_{23} + a_{31}\, a_{13}\, a_{23} + a_{11}\, a_{22}\, a_{33}$$
$$= -(s)\ (11)\ (-1) + (6)\ (-1)\ (-1) + s^2\ (s+6)$$

and

$$\det \{(sI-A)+bc^T\} = -a_{11}\, a_{32}\, a_{23} + a_{31}\, a_{12}\, a_{23} + a_{11}\, a_{22}\, a_{33}$$
$$= -s(11+7)(-1) + (6+5)(-1)(-1) + s^2(s+6)$$
$$= S^3 + 6s^2 + 18s + 11 \tag{17}$$

Therefore, the desired transfer function can be obtained by substituting (16) and (17) into (14).

That is,

$$1 + c^T(sI-A)^{-1}b = \frac{s^3 + 6s^2 + 18s + 11}{s^3 + 6s^2 + 11s + 6} \tag{18}$$

or

$$c^T(sI-A)^{-1}b = \frac{7s + 5}{s^3 + 6s^2 + 11s + 6} \tag{19}$$

as we expected.

1.4 Gain Evaluation

The authors of this report generalized the utilization of Grassmann algebra to the gain evaluation and found that the outgoing branches of a node can be

interpreted as a Grassmann vector, and the determinant of a cold system can be obtained by taking the exterior product of those vectors.

For illustration, consider the control system shown in Figure 1, which can be represented by the block diagram shown in Figure 2. The block diagram can be expressed by a matrix equation:

$$\begin{bmatrix} -1 & 0 & 0 & 0 & 0 \\ K_g & -1 & 0 & 0 & 0 \\ 0 & 1/(R_1+sL_1) & -1 & 0 & -1/(R_1+sL_1) \\ 0 & 0 & K_t/(Js+b) & -1 & 0 \\ 0 & 0 & 0 & K_m & -1 \end{bmatrix} \begin{bmatrix} I_f \\ E_g \\ I_a \\ \Omega_m \\ E_m \end{bmatrix} = \begin{bmatrix} -1/(R_f + sL_f) \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} E_f \tag{20}$$

To apply the Grassmann algebra to equation (20), we first code each element in the matrix equation by a double script symbol and obtain

$$\begin{bmatrix} a_{11} & 0 & 0 & 0 & 0 \\ a_{21} & a_{22} & 0 & 0 & 0 \\ 0 & a_{32} & a_{33} & 0 & a_{35} \\ 0 & 0 & a_{43} & a_{44} & 0 \\ 0 & 0 & 0 & a_{54} & a_{55} \end{bmatrix} \begin{bmatrix} I_f \\ E_g \\ I_a \\ \Omega_m \\ E_m \end{bmatrix} = \begin{bmatrix} \beta_1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} E_f \tag{20a}$$

where $\beta_1 = -1/(R_f + sL_f)$.

We can now draw the flow-graph in the Coates sense which is shown in Figure 3, either directly from the block diagram or from the matrix equation.

Let us consider the cold system first which means that the system is without input. Redraw Figure 3 by deleting all the incoming branches. Then we have

Figure 4. The determinant of the cold system can be found from the exterior product of the vectors formed by the outgoing branches of each node. In other words.

$$\Delta_c = (a_{11}, a_{21}) \wedge (a_{22}, a_{32}) \wedge (a_{33}, a_{43})) \wedge (a_{44}, a_{54}) \wedge (a_{35}, a_{55}) \tag{21}$$

In Grassmann's notation, we write

$$\Delta_c = (1,2) \wedge (2,3) \wedge (3,4) \wedge (4,5) \wedge (3,5)$$

Then the exterior product operations are performed subject to the Grassmann's rules shown in Section 1.2. We have

$$\begin{aligned} \Delta_c &= (12,13,23) \wedge (3,4) \wedge (4,5) \wedge (3,5) \\ &= (123,124,134,234) \wedge (4,5) \wedge (3,5) \\ &= (1234,1235,1245,1345,2345) \wedge (3,5) \\ &= (12345,12453) \end{aligned} \tag{22}$$

which means

$$a_{11}\ a_{22}\ a_{33}\ a_{44}\ a_{55} + a_{11}\ a_{22}\ a_{43}\ a_{54}\ a_{35}$$

After checking even or odd permutation, the sign of each item is decoded. Therefore, we have

$$\Delta_c = + a_{11}\ a_{22}\ a_{33}\ a_{44}\ a_{55} + a_{11}\ a_{22}\ a_{43}\ a_{54}\ a_{35} \tag{23}$$

Substituting the original values gives

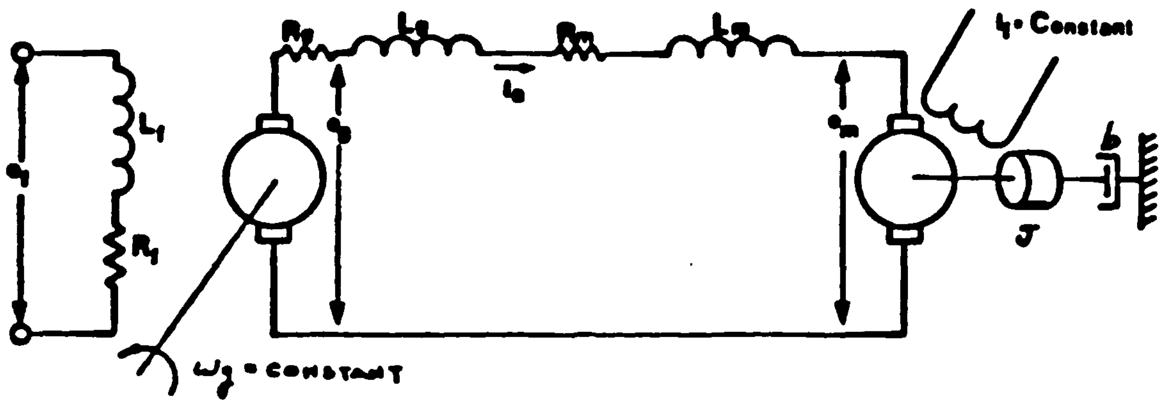


Figure 1. A Control System

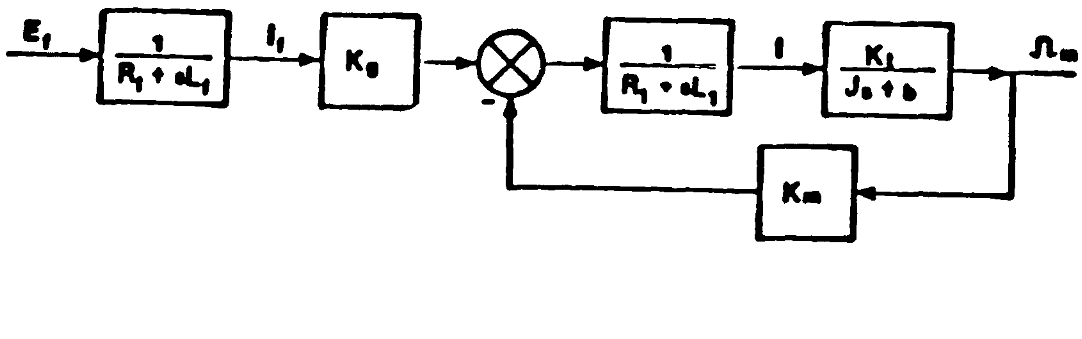


Fig. 2. Block Diagram of the Control System
($R_1 = R_g + R_m$, $L_1 = L_g + L_m$)

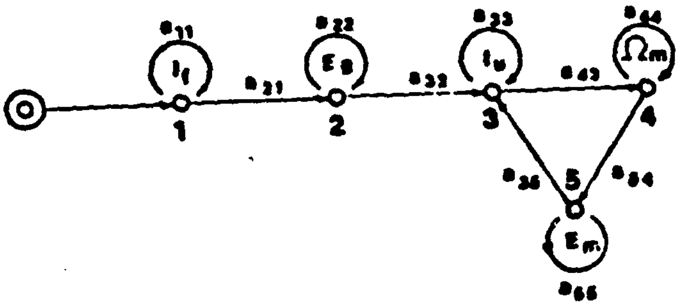


Figure 3. Flow Graph in Coates Sense

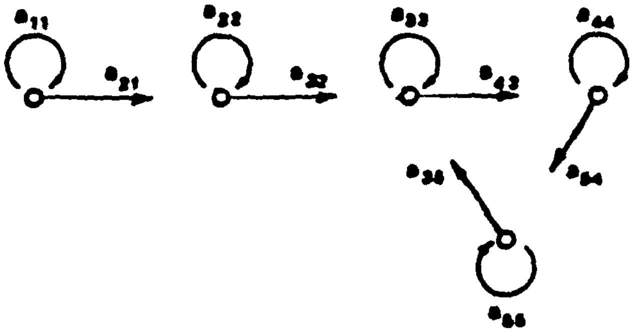


Figure 4. Only Outgoing Branches from Nodes Drawn

$$\Delta_c = (-1)(-1)(-1)(-1)(-1) + (-1)(-1)\left(\frac{K_T}{Js+b}\right)(K_m)\left(\frac{-1}{R_1+sL_1}\right)$$

$$= -1 - \frac{K_T\,K_m}{(Js + b)(R_1 + sL_1)} \qquad (24)$$

Which is the denominator of the gain.

If we are interested in the gain between say, Ω_m to E_f, we write the output equation as follows

$$y = [0, 0, 0, 1, 0]\ [I_f, E_g, I_a, \Omega_m, E_m]^T \qquad (25)$$

$$= c^T x$$

By using (14), we finally get the numerator of the gain

$$N = -\left(K_g \times \frac{1}{R_1+sL_1} \times \frac{K_T}{Js + b} \times \frac{-1}{R_f+sL_f} \times (-1)\right) \qquad (26)$$

Combining (24) and (26) yields the gain, $\Omega_m(s)/E_f(s)$.

As we have seen, the new method based on Grassmann's algebra is not only suitable for computers to deal with large scale systems, but also preserves symbols all the way. While Mason's formula frequently makes mistakes for large scale operations, and Coates' formula is only a different form of Cramer's rule, our approach is more accurate than that of Mason and more general than that of Coates.

For the simplicity of generalization, we need to formalize the steps from (20) to (26). Since any single input and single output (SISO) signal flow graph is

equivalent to a linear system of the form

$$Ax = bu \qquad (27)$$

$$y = c^T x. \qquad (28)$$

where A is the Coates graph matrix, u the input, y the output, b the column vector consists of all the negative gains from the input node to all other nodes in the flow graph, and c^T is the row vector of the form $[c_1, \ldots, c_i, \ldots, c_n]$, where c_i is the gain from the i^{th} state variable node to the output node y.

We have

$$x = A^{-1}bu,$$

and $y = c^T A^{-1} bu,$

the transfer function or the gain of the flow graph

$$G = c^T A^{-1} b = \Delta_n / \Delta_d. \qquad (29)$$

Compare the $c^T A^{-1} b$ with the $c^T(sI-A)^{-1}b$ in (14), we have

$$1 + c^T A^{-1} b = (\Delta_n + \Delta_d)/(\Delta_d) = \frac{\det(A + bc^T)}{\det A}$$

or

$$1 + G = \frac{\det(A + bc^T)}{\det A} \qquad (30)$$

For elaboration, examine the following example.

Example 1.

Evaluate the gain of the discrete signal flow graph shown in Figure 5.

Using matrix notations and the dummy variables α_1 and α_2, the flow graph is equivalent to the following linear system

$$\begin{bmatrix} -1 & -d_k & 0 \\ Z^{-1} & -1 & 0 \\ 1 & c_k & -1 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ u(k) \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix} r(k)$$

$$u(k) = [0\ 0\ 1] \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ u(k) \end{bmatrix}$$

Then, the cold system(flow graph without the input) is equivalent to the Coates graph shown in Figure 6.

The determinant Δ_d is found by evaluating

$$\begin{bmatrix} 1 & 1 & 0 \\ 2 & 2 & 0 \\ 3 & 3 & 3 \end{bmatrix}$$

symbolically

$$\begin{aligned} \Delta_d &= (\ 1\ 2\ 3\ ) - (\ 2\ 1\ 3\ ) \\ &= (-1)(-1)(-1) - (Z^{-1})(-d_k)(-1) \\ &= -(1+d_k Z^{-1}) \end{aligned}$$

To found the numerator Δ_n, we need to identify the b and c^T vectors

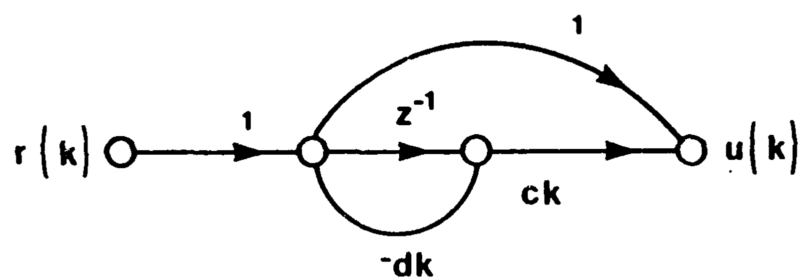


FIGURE 5

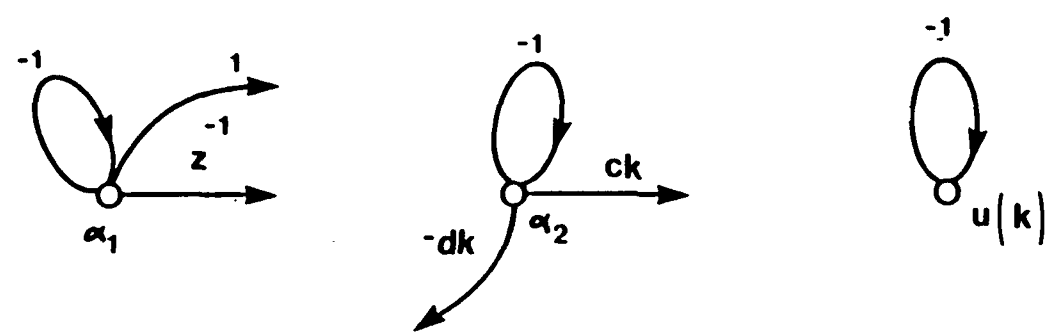


FIGURE 6

$$bc^T = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Therefore,

$$A + bc^T = \begin{bmatrix} -1 & -d_k & -1 \\ Z^{-1} & -1 & 0 \\ 1 & c_k & -1 \end{bmatrix}$$

which is evaluated symbolically by using the matrix

$$\begin{bmatrix} 1 & 1 & 1 \\ 2 & 2 & 0 \\ 3 & 3 & 3 \end{bmatrix}$$

Therefore,

$$\begin{aligned} \Delta_n + \Delta_d &= (1\ 2\ 3) - (2\ 1\ 3) + (2\ 3\ 1) - (3\ 2\ 1) \\ &= (-1)(-1)(-1)-(Z^{-1})(-d_k)(-1)+(Z^{-1})(c_k)(-1)-(1)(-1)(-1) \\ &= -(1+d_kZ^{-1})-(1+c_kZ^{-1}), \end{aligned}$$

$$\Delta_n = -(1+c_kZ^{-1}).$$

From (29), the gain

$$G(Z^{-1}) = \frac{u(k)}{r(k)} = \frac{\Delta_n}{\Delta_d} = \frac{1+c_kZ^{-1}}{1+d_kZ^{-1}}$$

The equations (29) and (30) will be generalized in the next chapter.

2.0 Symbolic Analysis of Multivariable Systems

This chapter is an extension to the last one. We will generalize all the SISO results to Multi-inputs and Multi-outputs (MIMO) cases.

2.1 Generalization of Grassmann Algebra

From the previous Chapter, for a SISO state equation, we have the result (14)

$$1 + c^T(sI-A)^{-1}b = \frac{\det\{(sI-A)+bc^T\}}{\det(sI-A)}$$

For an n inputs and m outputs system, the state equations are

$$\dot{x} = Ax + Bu$$

$$Y = C^Tx$$

The transfer function matrix

$$G(s) = C^T(sI-A)^{-1}B = [\ g_{ij}(s)\ ] = \begin{bmatrix} g_{11}(s) & g_{12}(s) & \cdots & g_{1m}(s) \\ g_{21}(s) & g_{22}(s) & \cdots & g_{2m}(s) \\ . & . & & . \\ . & . & & . \\ g_{n1}(s) & g_{n2}(s) & \cdots & g_{nm}(s) \end{bmatrix} \tag{31}$$

where

$$C^T = \begin{bmatrix} c_1^T \\ c_2^T \\ . \\ . \\ . \\ c_n^T \end{bmatrix} \quad \text{and} \quad B = [b_1,\ b_2,\ \ldots,\ b_m].$$

Therefore,

$$G(s) = \begin{bmatrix} c_1^T(sI-A)^{-1} \\ c_2^T(sI-A)^{-1} \\ . \\ . \\ c_n^T(sI-A)^{-1} \end{bmatrix} [\ b_1,\ b_2,\ \ldots,\ b_m\ ]$$

$$
= \begin{bmatrix} c_1^T(sI-A)^{-1}b_1 & c_1^T(sI-A)^{-1}b_2 & \cdots & c_1^T(sI-A)^{-1}b_m \\ c_2^T(sI-A)^{-1}b_1 & c_2^T(sI-A)^{-1}b_2 & \cdots & c_2^T(sI-A)^{-1}b_m \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ c_n^T(sI-A)^{-1}b_1 & \cdot & & c_n^T(sI-A)^{-1}b_m \end{bmatrix}
$$

Then, we have

$$
1 + g_{ij}(s) = 1 + c_i^T(sI-A)^{-1}b_j = \frac{\det\{(sI-A)+b_jc_i^T\}}{\det(sI-A)} \qquad (32)
$$

$$
= \Delta_{ij}/\Delta_d,
$$

where $\Delta_{ij} = \det\{(sI-A)+b_jc_j^T\} - \Delta_d$, (33a)

and $\Delta_d = \det(sI-A)$. (33b)

Comparing (32) and (33) with (14), it is found that we have reduced the evaluation of the n×m transfer matrix into the evaluation of nm SISO transfer functions.

Example 2.

To illustrate the generalization of Grassmann algebra, consider the RLC network shown in Figure 7. Let the voltages V_1 and V_2 be the inputs and the currents I_1 and I_2 be the outputs. The loop equations are

$$-V_1 + I_1R + (I_1-I_2)/sC = 0,$$

and $$V_2 + I_2(R+sL) + (I_2-I_1)/sC = 0.$$

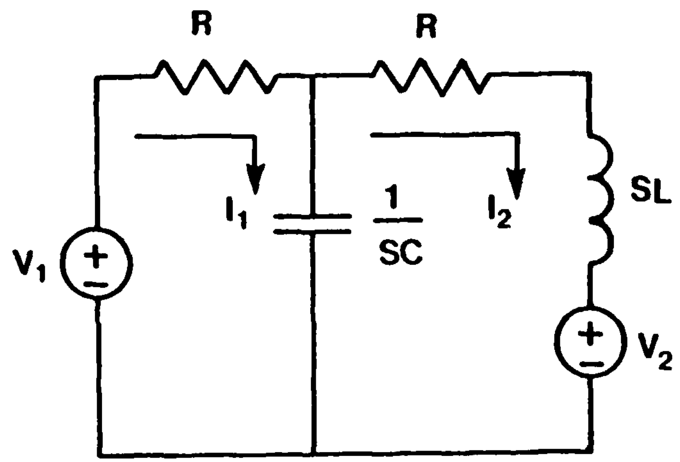


FIGURE 7

rearranging, we have

$$\begin{bmatrix} R+(1/sC) & -(1/sC) \\ -(1/sC) & R+(1/sC)+sL \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix}$$

$$Y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}$$

The transfer function matrix is given by

$$G(s) = [\ g_{ij}(s)\ ] = \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix}$$

and the $g_{ij}(s)$ is given by (32)

$$\Delta_d = \det \begin{bmatrix} R+(1/sC) & -(1/sC) \\ -(1/sC) & R+(1/sC)+sL \end{bmatrix} = \frac{R+(R+sL)(1+RCs)}{sC}$$

$$b_1c_1^T = \begin{bmatrix} 1 \\ 0 \end{bmatrix} [1\ \ 0] = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

Therefore,

$$\Delta_{11} + \Delta_d = \det \begin{bmatrix} 1+R+(1/sC) & -(1/sC) \\ -(1/sC) & R+(1/sC)+sL \end{bmatrix}$$

$$\Delta_{11} = \frac{(1+RCs)(s^2LC) - 1}{(sC)^2}$$

$$g_{11} = \Delta_{11}/\Delta_d$$

$$= \frac{(1+RCs)(s^2LC) - 1}{(sC)[\ R+(R+sL)(1+RCs)\ ]}$$

To evaluate the g_{12}, we use

$$b_2 c_1^T = \begin{bmatrix} 0 \\ -1 \end{bmatrix} [1\ 0] = \begin{bmatrix} 0 & 0 \\ -1 & 0 \end{bmatrix}$$

$$\Delta_{12} + \Delta_d = \det \begin{bmatrix} R+(1/sC) & -(1/sC) \\ -(1+(1/sC)) & R+(1/sC)+sL \end{bmatrix}$$

$$\Delta_{12} = -(1/sC),$$

Therefore

$$g_{12} = \Delta_{12}/\Delta_d = \frac{-1}{R+(R+sL)(1+RCs)}$$

Similarly, we obtain

$$\Delta_{21} = 1/sC,$$

and

$$\Delta_{22} = \frac{-(1+RCs)}{sC}$$

The transfer matrix is

$$G(s) = \begin{bmatrix} \dfrac{(1+RCs)(s^2LC)-1}{(sC)[R+(R+sL)(1+RCs)]} & \dfrac{-1}{R+(R+sL)(1+RCs)} \\ \dfrac{1}{R+(R+sL)(1+RCs)} & \dfrac{-(1+RCs)}{R+(R+sL)(1+RCs)} \end{bmatrix}$$

Example 3.

Consider the state equations

$$\dot{x} = Ax + Bu$$

$$Y = C^T x.$$

where

$$A = \begin{bmatrix} -4 & 1 & 0 & 0 & 0 \\ -5 & 0 & 1 & 0 & 0 \\ -2 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & -4 & 1 \\ -2 & 0 & 0 & -4 & 0 \end{bmatrix},\ B = \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ -1 & -1 \\ 0 & 0 \end{bmatrix},\ x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}.$$

$$C^T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix},\ u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix},\ Y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}.$$

The transfer function matrix is given by

$$G(s) = [\ g_{ij}(s)\ ] = \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix}$$

and the $g_{ij}(s)$ is given by (32).

When the $b_j c_i^T$'s are expanded and added up with the (sI-A) matrix, it is found that only $(sI-A+b_2c_2^T)$ needs a different symbolic notation, all the other matrices have the same symbolic notation as that of the (sI-A) matrix.

For the (sI-A) type, the follow symbolic matrix is used

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 2 & 2 & 2 & 0 & 0 \\ 3 & 0 & 3 & 0 & 0 \\ 4 & 0 & 0 & 4 & 0 \\ 5 & 0 & 0 & 5 & 5 \end{bmatrix}$$

and the determinant is given symbolically as

$$(1\ 2\ 3\ 4\ 5)-(1\ 2\ 3\ 5\ 4)-(2\ 1\ 3\ 4\ 5)+(2\ 1\ 3\ 5\ 4)+(3\ 1\ 2\ 4\ 5)-(3\ 1\ 2\ 5\ 4)$$

For the $(sI-A+b_2c_2^T)$, we must use the symbolic matrix

$$\begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 2 & 2 & 2 & 0 & 0 \\ 3 & 0 & 3 & 0 & 0 \\ 4 & 0 & 0 & 4 & 4 \\ 5 & 0 & 0 & 5 & 5 \end{bmatrix}$$

and the determinant is given symbolically as

$$(1\ 2\ 3\ 4\ 5)-(1\ 2\ 3\ 5\ 4)-(2\ 1\ 3\ 4\ 5)+(2\ 1\ 3\ 5\ 4)+$$
$$(3\ 1\ 2\ 4\ 5)-(3\ 1\ 2\ 5\ 4)-(4\ 2\ 3\ 1\ 5)+(5\ 2\ 3\ 1\ 4)$$

For illustration, we find det(sI-A).

$$(sI-A) = \begin{bmatrix} s+4 & -1 & 0 & 0 & 0 \\ 5 & s & -1 & 0 & 0 \\ 2 & 0 & s & 0 & 0 \\ 1 & 0 & 0 & s+4 & -1 \\ 2 & 0 & 0 & 4 & s \end{bmatrix}$$

on substitution,

$$\det(sI-A) = \Delta_d = (s+4)(s)(s)(s+4)(s)-(s+4)(s)(s)(4)(-1)-$$
$$(5)(-1)(s)(s+4)(s)+(5)(-1)(s)(4)(-1)+$$
$$(2)(-1)(-1)(s+4)(s)-(2)(-1)(-1)(4)(-1)$$
$$= s^5 + 8s^4 + 25s^3 + 38s^2 + 28s + 8$$
$$= (s+1)^2 (s+2)^3$$

For the $\det\{(sI-A)+b_2c_2^T\}$,

$$(sI-A)+b_2c_2^T = \begin{bmatrix} s+4 & -1 & 0 & 1 & 0 \\ 5 & s & -1 & 0 & 0 \\ 2 & 0 & s & 0 & 0 \\ 1 & 0 & 0 & s+3 & -1 \\ 2 & 0 & 0 & 4 & s \end{bmatrix}$$

on substitution,

$$\det\{(sI-A)+b_2c_2^T\} = (s+4)(s)(s)(s+3)(s)-(s+4)(s)(s)(4)(-1)-$$
$$(5)(-1)(s)(s+3)(s)+(5)(-1)(s)(4)(-1)+$$
$$(2)(-1)(-1)(s+3)(s)-(2)(s)(s)(1)(-1)-$$
$$(1)(s)(s)(1)(s)+(2)(s)(s)(1)(-1)$$
$$= s^5 + 7s^4 + 20s^3 + 31s^2 + 26s + 8$$

$$\Delta_{22} = \det\{(sI-A)+b_2c_2^T\} - \det\{(sI-A)\}$$
$$= -s\ (\ s^3 + 5s^2 + 7s + 2\ )$$
$$= -s\ (\ s + 2\ )\ (\ s^2 + 3s + 1\ )$$
$$g_{22}(s) = \Delta_{22}/\Delta_d$$
$$= \frac{-(s)(s+2)(s^2+3s+1)}{(s+1)^2(s+2)^3} = \frac{-s(s^2+3s+1)}{(s+1)^2(s+2)^3}$$

using the different $(sI-A)+b_jc_i^T$ matrices, we have

$$\det(sI-A+b_1c_1^T) = s^5 + 8s^4 + 25s^3 + 38s^2 + 28s + 8$$
$$\det(sI-A+b_2c_1^T) = s^5 + 9s^4 + 29s^3 + 42s^2 + 28s + 8$$
$$\det(sI-A+b_1c_2^T) = s^5 + 7s^4 + 21s^3 + 33s^2 + 26s + 8$$

Then

$$\Delta_{11} = 0$$
$$\Delta_{12} = s^4 + 4s^3 + 4s^2 = s^2\ (s + 2)$$
$$\Delta_{21} = -s(s^3 + 4s^2 + 5s + 2) = -s(s+1)^2(s+2)$$

and the transfer function matrix is given by

$$G(s) = \begin{bmatrix} 0 & \dfrac{s^2}{(s+1)^2(s+2)} \\ \dfrac{-s}{(s+2)^2} & \dfrac{-s(s^2+3s+1)}{(s+1)^2(s+2)} \end{bmatrix}$$

2.2 Multivariable Signal Flow Graphs

The general form of the linear equations of MIMO signal flow graph is

$$Ax = BU$$
$$Y = C^Tx.$$

where A is the Coates graph matrix, U the input matrix, Y the outputs, B is a matrix with the i^{th} column the negative gains from the i^{th} input to all other nodes in the flow graph, and C^T is a matrix with the j^{th} row the gains from the state variable nodes to the j^{th} output node y_j of Y.

Which implies that the transfer function matrix

$$G(s) = [\, g_{ij}(s) \,] = C^T A^{-1} B$$

$$= \begin{bmatrix} c_1^T \\ \cdot \\ \cdot \\ \cdot \\ c_n^T \end{bmatrix} A^{-1} [\, b_1, b_2, \ldots, b_m \,]$$

$$= \begin{bmatrix} c_1^T A^{-1} b_1 & \ldots & c_1^T A^{-1} b_m \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ c_n^T A^{-1} b_1 & \ldots & c_n^T A^{-1} b_m \end{bmatrix}$$

that is

$$g_{ij} = c_i^T A^{-1} b_j = \Delta_{ij}/\Delta_d \qquad (34a)$$

or,

$$1 + g_{ij} = \frac{(\, \Delta_{ij} + \Delta_d \,)}{\Delta_d} = \frac{\det(\, A + b_j c_i^T \,)}{\det A} \qquad (34b)$$

Comparing (34a) and (34b) with (29) and (30) respectively, we will find that we have reduced the symbolic gain evaluation of an m inputs and n outputs signal flow graph into the gains evaluation of nm single input and single output signal flow graph.

Example 4.

Evaluate the transfer function matrix of the discrete signal flow graph shown in Figure 8.

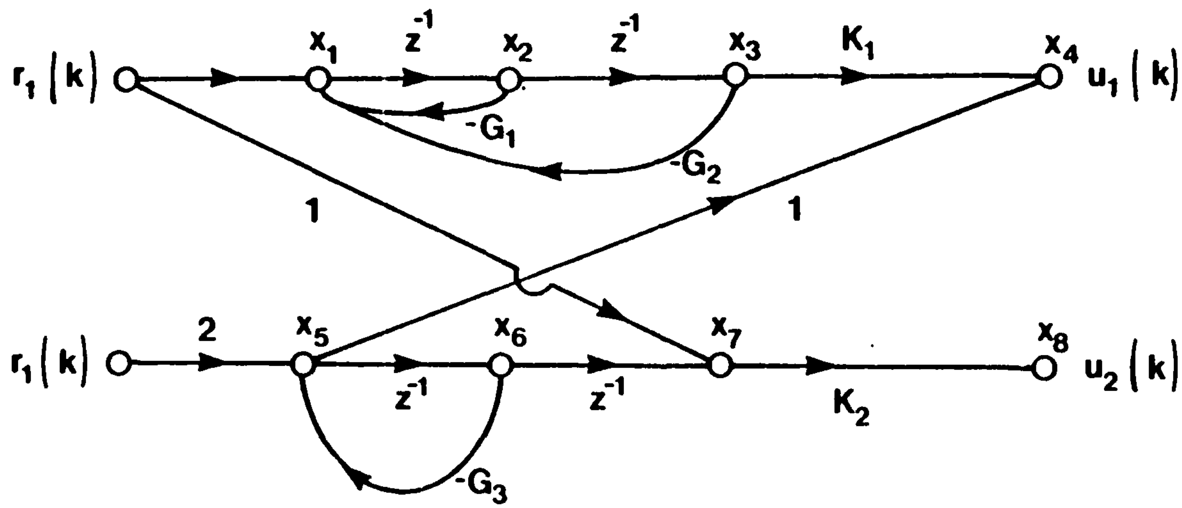


FIGURE 8

The 2-inputs and 2-outputs signal flow graph is equivalent to the following linear system

$$\begin{bmatrix} -1 & -G_1 & -G_2 & 0 & 0 & 0 & 0 & 0 \\ Z^{-1} & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Z^{-1} & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & K_1 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -G_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & Z^{-1} & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & Z^{-1} & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & K_2 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & -2 \\ -1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} r_1(k) \\ r_2(k) \end{bmatrix}$$

$$\begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\Delta_d = \det \begin{bmatrix} -1 & -G_1 & -G_2 & 0 & 0 & 0 & 0 & 0 \\ Z^{-1} & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Z^{-1} & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & K_1 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -G_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & Z^{-1} & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & Z^{-1} & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & K_2 & -1 \end{bmatrix}$$

symbolically,

$$\Delta_d = \det \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 4 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 6 & 6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 7 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 8 & 8 \end{bmatrix}$$

$$= (1\ 2\ 3\ 4\ 5\ 6\ 7\ 8) - (1\ 2\ 3\ 4\ 6\ 5\ 7\ 8) - (2\ 1\ 3\ 4\ 5\ 6\ 7\ 8) + (2\ 1\ 3\ 4\ 6\ 5\ 7\ 8) + (2\ 3\ 1\ 4\ 5\ 6\ 7\ 8) - (2\ 3\ 1\ 4\ 6\ 5\ 7\ 8)$$

$$= (-1)(-1)(-1)(-1)(-1)(-1)(-1)(-1) - (-1)(-1)(-1)(-1)(Z^{-1})(-G_3)(-1)(-1) - (Z^{-1})(-G_1)(-1)(-1)(-1)(-1)(-1)(-1) + (Z^{-1})(-G_1)(-1)(-1)(Z^{-1})(-G_3)(-1)(-1) + (Z^{-1})(Z^{-1})(-G_2)(-1)(-1)(-1)(-1)(-1) + (Z^{-1})(Z^{-1})(-G_2)(-1)(Z^{-1})(-G_3)(-1)(-1)$$

$$= 1 + G_3Z^{-1} + G_1Z^{-1} + G_1G_3Z^{-2} + G_2Z^{-2} + G_2G_3Z^{-3}$$

$$= (1+G_1Z^{-1}+G_2Z^{-2})(1+G_3Z^{-1})$$

$$\Delta_{11} + \Delta_d = \begin{bmatrix} -1 & -G_1 & -G_2 & -1 & 0 & 0 & 0 & 0 \\ Z^{-1} & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Z^{-1} & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & K_1 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -G_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & Z^{-1} & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & Z^{-1} & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & K_2 & -1 \end{bmatrix}$$

Symbolically,

$$\Delta_{11} + \Delta_d = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 4 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 & 5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 6 & 6 & 0 & 0 \\ 0 & 0 & 0 & 7 & 0 & 7 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 8 & 8 \end{bmatrix}$$

$$= (1\ 2\ 3\ 4\ 5\ 6\ 7\ 8)-(1\ 2\ 3\ 4\ 6\ 5\ 7\ 8)-(2\ 1\ 3\ 4\ 5\ 6\ 7\ 8)+$$

$$(2\ 1\ 3\ 4\ 6\ 5\ 7\ 8)+(2\ 3\ 1\ 4\ 5\ 6\ 7\ 8)-(2\ 3\ 1\ 4\ 6\ 5\ 7\ 8)-$$

$$(2\ 3\ 4\ 1\ 5\ 6\ 7\ 8)+(2\ 3\ 4\ 1\ 6\ 5\ 7\ 8)$$

$$= \Delta_d + K_1 Z^{-2}(1+G_3 Z^{-1})$$

Therefore $\Delta_{11} = K_1 Z^{-2}(1+G_3 Z^{-1})$

$$g_{11} = \frac{\Delta_{11}}{\Delta_d} = \frac{K_1 Z^{-2}}{1+G_1 Z^{-1}+G_2 Z^{-2}}$$

Similarly, we obtain

$$\Delta_{12} = K_2(1+G_1 Z^{-1}+G_2 Z^{-2})(1+G_3 Z^{-1})$$

$$\Delta_{21} = 2(1+G_1 Z^{-1}+G_2 Z^{-2})$$

$$\Delta_{22} = (2K_2 Z^{-2})(1+G_1 Z^{-1}+G_2 Z^{-2})$$

The transfer matrix

$$G = \begin{bmatrix} \dfrac{K_1 Z^{-2}}{1+G_1 Z^{-1}+G_2 Z^{-2}} & K_2 \\ \\ \dfrac{2}{1+G_3 Z^{-1}} & \dfrac{2K_2 Z^{-2}}{1+G_3 Z^{-1}} \end{bmatrix}$$

Furthermore, we show the superposition technique for tackling MIMO system problems here. Using the superposition property of linear systems, different components of the transfer function matrix can be found by the reduced flow graphs in Figure 8.

From Figure 9(a), the $g_{11}(Z^{-1})$ is found by use

$$\Delta_d = \det \begin{bmatrix} -1 & -G_1 & -G_2 & 0 \\ Z^{-1} & -1 & 0 & 0 \\ 0 & Z^{-1} & -1 & 0 \\ 0 & 0 & K_1 & -1 \end{bmatrix}$$

and is evaluated symbolically by using

$$\begin{bmatrix} 1 & 1 & 1 & 0 \\ 2 & 2 & 0 & 0 \\ 0 & 3 & 3 & 0 \\ 0 & 0 & 4 & 4 \end{bmatrix}$$

$$\begin{aligned} \Delta_d &= (1\ 2\ 3\ 4)-(2\ 1\ 3\ 4)+(2\ 3\ 1\ 4) \\ &= (-1)(-1)(-1)(-1)-(Z^{-1})(-G_1)(-1)(-1)+(Z^{-1})(Z^{-1})(-G_2)(-1) \\ &= 1 + G_1 Z^{-1} + G_2 Z^{-2} \end{aligned}$$

with

$$b^T = [-1\ 0\ 0\ 0] \text{ and } c^T = [0\ 0\ 0\ 1]$$

$\Delta_{11} + \Delta_d$ is evaluated symbolically by using

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 0 & 0 \\ 0 & 3 & 3 & 0 \\ 0 & 0 & 4 & 4 \end{bmatrix}$$

$$\begin{aligned} \Delta_{11} + \Delta_d &= (1\ 2\ 3\ 4)-(2\ 1\ 3\ 4)+(2\ 3\ 1\ 4)-(2\ 3\ 4\ 1) \\ &= (-1)(-1)(-1)(-1)-(Z^{-1})(-G_1)(-1)(-1)+(Z^{-1})(Z^{-1})(-G_2)(-1)- \\ &\qquad (Z^{-1})(Z^{-1})(K_1)(-1) \\ &= \Delta_d + K_1 Z^{-2} \end{aligned}$$

$$\Delta_{11} = K_1 Z^{-2}$$

$$g_{11}(Z^{-1}) = \frac{\Delta_{11}}{\Delta_d} = \frac{K_1 Z^{-2}}{1+G_1 Z^{-1}+G_2 Z^{-2}}$$

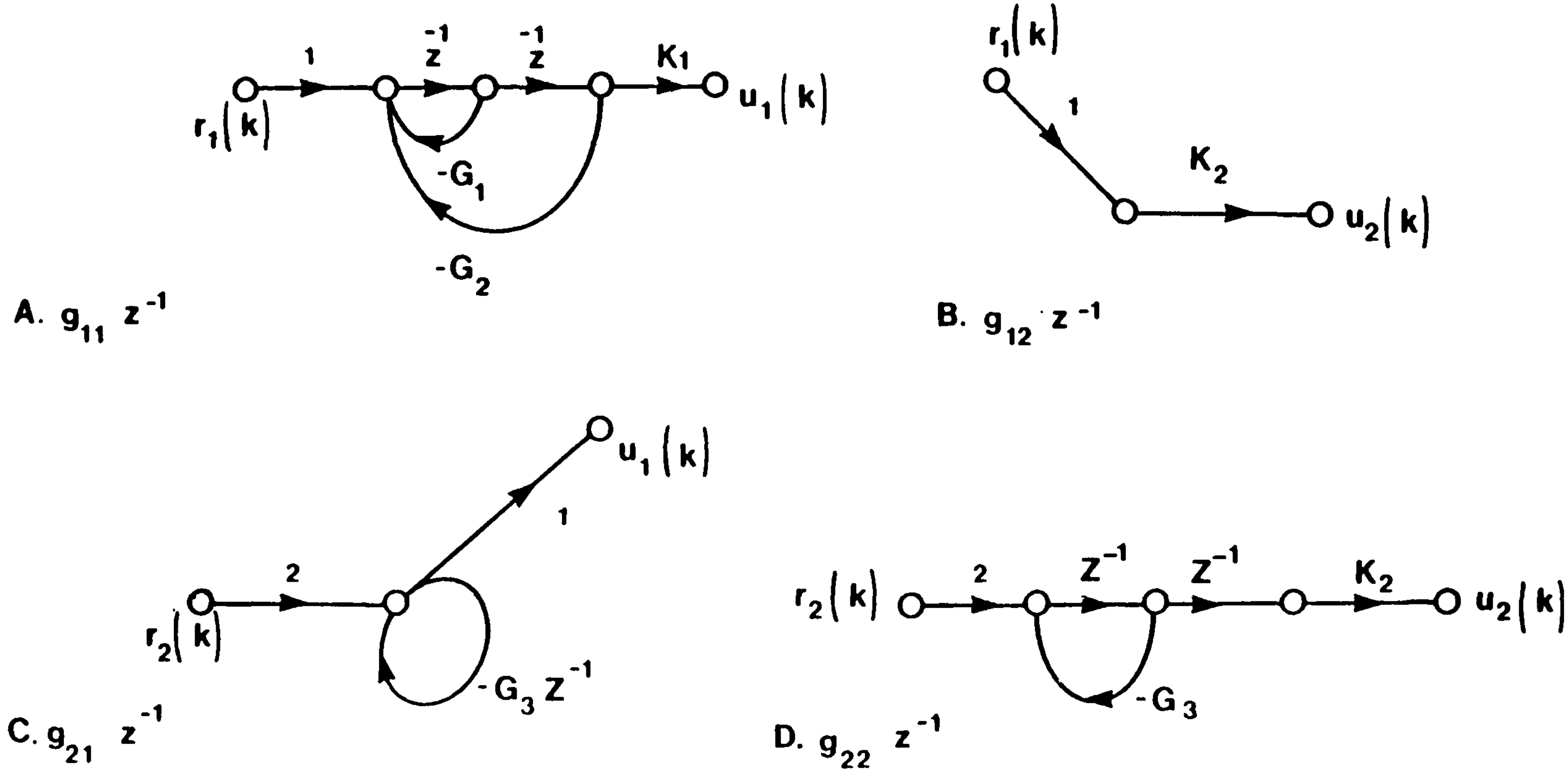

1
z^{-1}
z^{-1}
K_1
$r_1(k)$
$u_1(k)$
$-G_1$
$-G_2$
A. $g_{11}\ z^{-1}$
$r_1(k)$
1
K_2
$u_2(k)$
B. $g_{12}\ z^{-1}$
$u_1(k)$
1
2
$r_2(k)$
$-G_3 Z^{-1}$
C. $g_{21}\ z^{-1}$
$r_2(k)$
2
Z^{-1}
Z^{-1}
K_2
$u_2(k)$
$-G_3$
D. $g_{22}\ z^{-1}$

FIGURE 9

Similarly, the g_{12} is evaluated by using a 2×2 matrix, the g_{21} by a 2×2 and the g_{22} by a 4×4. Instead of using an 8×8 matrix, the use of the superposition property of linear system simplifies the gain evaluation to a certain extend. This method has some similarity with the aggregation methods in control theories and worth further investigation.

2.3 Flow Graph Decomposition

When the number of nodes in a flow graph is large, the symbolic evaluation of the determinant of the graph becomes intractable. A recursive decomposing technique is given here.

Given any matrix, we can partition it into the following matrix block form

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

where A is partitioned in such a way that it can be handled by the Grassmann algebra with no problems.

Then, as (12),

$$\det \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \det A \det(D - CA^{-1}B)$$

Compare the matrix $CA^{-1}B$ with the $C^T(sI-A)^{-1}B$ in (31), we have

$$G = [\ g_{ij}\ ] = CA^{-1}B = \begin{bmatrix} c_1^T \\ c_2^T \\ \cdot \\ \cdot \\ \cdot \\ c_n^T \end{bmatrix} A^{-1} [\ b_1, \ldots, b_m\ ]$$

$$= \begin{bmatrix} c_1A^{-1}b_1 & . & . & . & c_1A^{-1}b_m \\ c_2A^{-1}b_1 & . & . & . & c_2A^{-1}b_m \\ . & & & & . \\ . & & & & . \\ c_nA^{-1}b_1 & . & . & . & c_nA^{-1}b_m \end{bmatrix}$$

That is,

$$1 + g_{ij} = \frac{\det(\ A+b_jc_i^T\ )}{\det A} \tag{35}$$

(35) implies that we need to solve a MIMO system. In terms of matrix flow graph, let

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

the matrix flow graph is shown in Figure 10(a). When it is partitioned, Figure 10(b) is obtained. As far as the determinants concern, we can decompose Figure 10(b) into Figure 10(c), which is also equivalent to Figure 10(d).

Since we partitioned in such a way that A_{11} is manageable, there is no problem in evaluation $A_{21}A_{11}{}^{-1}A_{12}$. If the dimension of the matrix $(A_{22}-A_{21}A_{11}{}^{-1}A_{12})$ is still too large to manipulate, we can repeat the algorithm again.

Example 3.

Consider Figure 11(a), evaluate the symbolic gain of this flow graph.

The matrix flow graph notation is shown in Figure 11(b) and the Coates graph matrix is

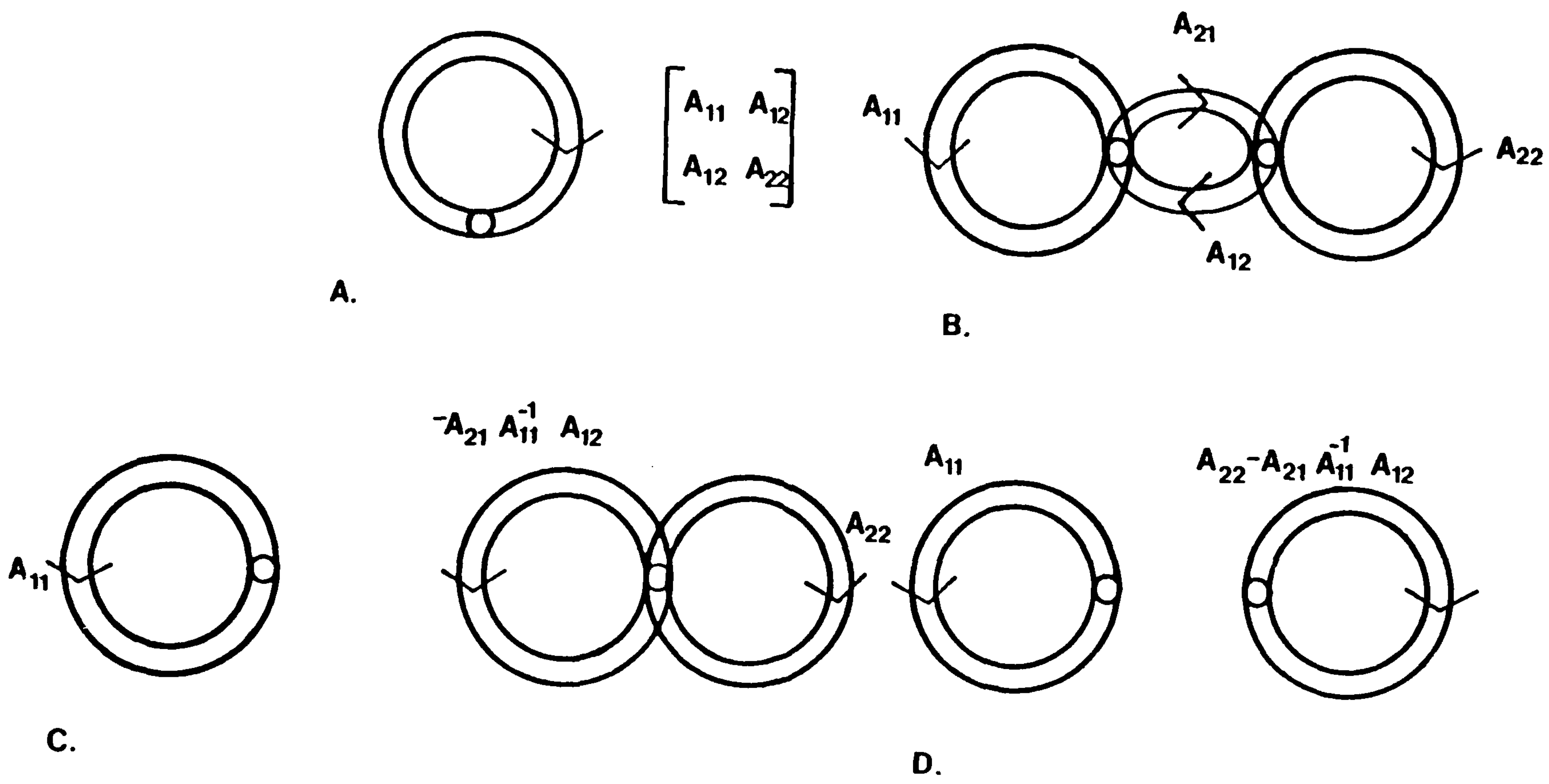
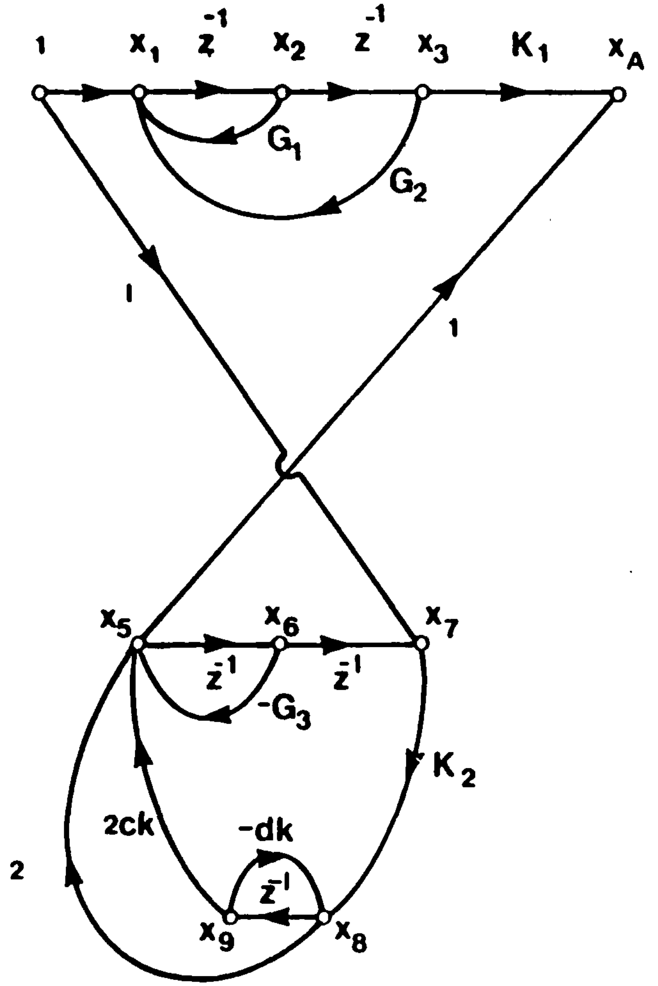
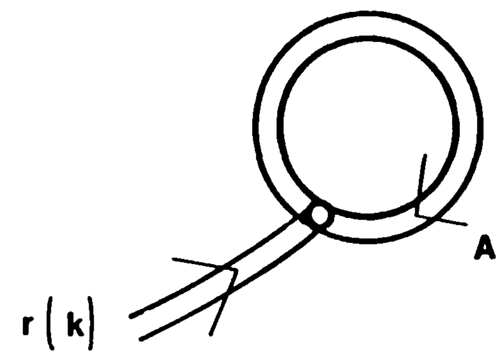

A.
$\begin{bmatrix} A_{11} & A_{12} \\ A_{12} & A_{22} \end{bmatrix}$
B.
A_{11}
A_{21}
A_{12}
A_{22}
C.
A_{11}
D.
$-A_{21} A_{11}^{-1} A_{12}$
A_{22}
A_{11}
$A_{22} - A_{21} A_{11}^{-1} A_{12}$

FIGURE 10



A.

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$



B.

FIGURE 11

$$
Ax = \begin{bmatrix} -1 & -G_1 & -G_2 & -1 & 0 & 0 & 0 & 0 & 0 \\ Z^{-1} & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Z^{-1} & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & K_1 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -G_3 & 0 & 2 & 2c_k \\ 0 & 0 & 0 & 0 & Z^{-1} & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & Z^{-1} & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & K_2 & -1 & -d_k \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & Z^{-1} & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ 0 \\ 0 \end{bmatrix} r(k)
$$

$$u(k) = [\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ ]\ x$$

To find the determinant Δ_d, we partition the matrix into the form

$$\begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix}$$

where

$$
A_1 = \begin{bmatrix} -1 & -G_1 & -G_2 & 0 \\ Z^{-1} & -1 & 0 & 0 \\ 0 & Z^{-1} & -1 & 0 \\ 0 & 0 & K_1 & -1 \end{bmatrix} \quad \text{and} \quad D_1 = \begin{bmatrix} -1 & -G_3 & 0 & 2 & 2c_k \\ Z^{-1} & -1 & 0 & 0 & 0 \\ 0 & Z^{-1} & -1 & 0 & 0 \\ 0 & 0 & K_2 & -1 & -d_k \\ 0 & 0 & 0 & Z^{-1} & -1 \end{bmatrix}
$$

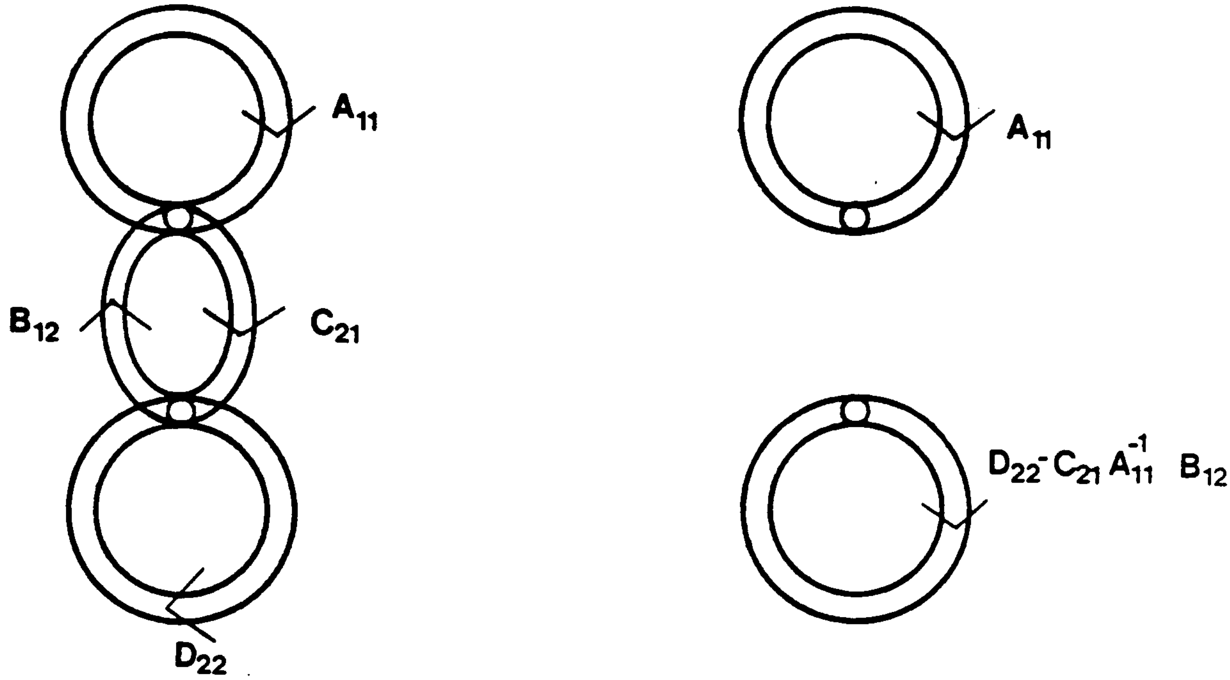
When the nodes (x_1, x_2, x_3, x_4) are grouped together as one matrix node and (x_5, x_6, x_7, x_8, x_9) as another, we have the matrix flow graph shown in Figure 12(a). As far as the determinants concern, we can decompose the matrix flow graph into Figure 12(b). Then

$$\Delta_d = \det \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix}$$

$$= \det A_1 \det(D_1 - C_1 A_1^{-1} B_1)$$

Since $C_1 = [\ 0\ ]$

$$\Delta_d = \det A_1 \det D_1$$



A. B.

FIGURE 12

The determinants are evaluated symbolically by using

$$\begin{bmatrix} 1 & 1 & 1 & 0 \\ 2 & 2 & 0 & 0 \\ 0 & 3 & 3 & 0 \\ 0 & 0 & 4 & 4 \end{bmatrix} \text{ and } \begin{bmatrix} 1 & 1 & 0 & 1 & 1 \\ 2 & 2 & 0 & 0 & 0 \\ 0 & 3 & 3 & 0 & 0 \\ 0 & 0 & 4 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \end{bmatrix}$$

$$\Delta_d = (1+G_1Z^{-1}+G_2z^{-2}) \; [\; 2K_2Z^{-2}(1+c_kZ^{-1}) - (1+G_3Z^{-1})(1+d_kZ^{-1}) \;]$$

using the b and c^T, the determinant ($\Delta_n + \Delta_d$) is found by the matrix

$$\begin{bmatrix} -1 & -G_1 & -G_2 & -1 & 0 & 0 & 0 & 0 & 0 \\ Z^{-1} & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & Z^{-1} & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & K_1 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & -G_3 & 0 & 2 & 2c_k \\ 0 & 0 & 0 & 0 & Z^{-1} & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & Z^{-1} & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & K_2 & -1 & -d_k \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & Z^{-1} & -1 \end{bmatrix}$$

Partition into the form

$$\begin{bmatrix} A_1' & B_1' \\ C_1' & D_1' \end{bmatrix}$$

where

$$A_1' = \begin{bmatrix} -1 & -G_1 & -G_2 & -1 \\ Z^{-1} & -1 & 0 & 0 \\ 0 & Z^{-1} & -1 & 0 \\ 0 & 0 & K_1 & -1 \end{bmatrix} \text{ and } D_1' = \begin{bmatrix} -1 & -G_3 & 0 & 2 & 2c_k \\ Z^{-1} & -1 & 0 & 0 & 0 \\ 0 & Z^{-1} & -1 & 0 & 0 \\ 0 & 0 & K_2 & -1 & -d_k \\ 0 & 0 & 0 & Z^{-1} & -1 \end{bmatrix}$$

$$\Delta_n + \Delta_d = \det \begin{bmatrix} A_1' & B_1' \\ C_1' & D_1' \end{bmatrix}$$

$$= \det A_1' \det(D_1' - C_1'A_1'^{-1}B_1')$$

Now, $\det A_1' = 1 + G_1Z^{-1} + G_2Z^{-2} + K_1Z^{-2}$

$$C_1'A_1'^{-1}B_1' = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ g_{31} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Where

$$g_{31} = \frac{1+G_1Z^{-1}+G_2Z^{-2}}{1+G_1Z^{-1}+G_2Z^{-2}+K_1Z^{-2}}$$

Therefore,

$$D_1' - C_1'A_1'^{-1}B_1' = \begin{bmatrix} -1 & -G_3 & 0 & 2 & 2c_k \\ Z^{-1} & -1 & 0 & 0 & 0 \\ -g_{31} & Z^{-1} & -1 & 0 & 0 \\ 0 & 0 & K_2 & -1 & -d_k \\ 0 & 0 & 0 & Z^{-1} & -1 \end{bmatrix}$$

$$\det(D_1'-C_1'A_1'^{-1}B_1') = 2K_2Z^{-2}(1+c_kZ^{-1})-(1+d_kZ^{-1})(1+G_3Z^{-1})- g_{31}(2K_2)(1+c_kZ^{-1})$$

$$\Delta_n = 2K_2Z^{-1}(1+c_kZ^{-1})[\ K_1Z^{-3}-(1+G_1Z^{-1}+G_2Z^{-2})\ ]-K_1Z^{-2}(1+d_kZ^{-1})(1+G_3Z^{-1})$$

Then, we have

$$\frac{u(k)}{r(k)} = \frac{\Delta_n}{\Delta_d} = \frac{2K_2Z^{-1}(1+c_kZ^{-1})[K_1Z^{-3}-(1+G_1Z^{-1}+G_2Z^{-2})]-K_1Z^{-2}(1+d_kZ^{-1})(1+G_3Z^{-1})}{(1+G_1Z^{-1}+G_2Z^{-2})[2K_2Z^{-2}(1+c_kZ^{-1})-(1+G_3Z^{-1})(1+d_kZ^{-1})]}$$

To illustrate the recursive nature of the algorithm, we apply it to the evaluation of the determinant of the matrices D_1 and $(D_1-C_1A_1^{-1}B_1)$. Partition the D_1 matrix into

$$\begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix}$$

where

$$A_2 = \begin{bmatrix} -1 & -G_3 & 0 \\ Z^{-1} & -1 & 0 \\ 0 & Z^{-1} & -1 \end{bmatrix} \quad \text{and} \quad D_2 = \begin{bmatrix} -1 & -d_k \\ Z^{-1} & -1 \end{bmatrix}$$

$$\det D_1 = \det A_2 \det(D_2 - C_2A_2^{-1}B_2)$$

$$\det A_2 = -1 - G_3Z^{-1}$$

$$C_2A_2^{-1}B_2 = \begin{bmatrix} g_{11} & g_{12} \\ 0 & 0 \end{bmatrix}$$

where

$$g_{11} = \frac{2K_2Z^{-2}}{-(1+G_3Z^{-1})}$$

$$g_{12} = \frac{2K_2c_kZ^{-2}}{-(1+G_3Z^{-1})}$$

$$\det(D_2-C_2A_2^{-1}B_2) = \det \begin{bmatrix} -1-g_{11} & -d_k-g_{12} \\ Z^{-1} & -1 \end{bmatrix}$$

$$= (1+g_{11})+(d_k+g_{12})Z^{-1}$$

$$= \frac{2K_2Z^{-2}(1+c_kZ^{-1}) - (1+G_3Z^{-1})(1+d_kZ^{-1})}{-(1+G_3Z^{-1})}$$

$$\det(A_2)\det(D_2-C_2A_2^{-1}B_2) = 2K_2Z^{-2}(1+c_kZ^{-1})-(1+G_3Z^{-1})(1+d_kZ^{-1})$$

Partition the matrix $(D_1'-C_1'A_1'^{-1}B_1')$ into

$$\begin{bmatrix} A_2' & B_2' \\ C_2' & D_2' \end{bmatrix}$$

where

$$A_2' = \begin{bmatrix} -1 & -G_3 & 0 \\ Z^{-1} & -1 & 0 \\ -g_{31} & Z^{-1} & -1 \end{bmatrix} \quad \text{and} \quad D_2' = \begin{bmatrix} -1 & -d_k \\ Z^{-1} & -1 \end{bmatrix}$$

$$\det A_2' = -1 - G_3Z^{-1}$$

$$C_2'A_2'^{-1}B_2' = \begin{bmatrix} g_{11}' & g_{12}' \\ 0 & 0 \end{bmatrix}$$

$$g_{11}' = \frac{2K_2(-g_{31}+Z^{-2})}{-(1+G_3Z^{-1})}$$

$$g_{12}' = \frac{2K_2c_k(-g_{31}+Z^{-2})}{-(1+G_3Z^{-1})}$$

$$\det(D_2'-C_2'A_2'^{-1}B_2') = \det \begin{bmatrix} -1-g_{11}' & -d_k-g_{12}' \\ Z^{-1} & -1 \end{bmatrix}$$

$$= (1+g_{11}')+(d_k+g_{12})Z^{-1}$$

$$= \frac{2K_2Z^{-2}(1+c_kZ^{-1})-(1+d_kZ^{-1})(1+G_3Z^{-1})-g_{31}(2K_2)(1+c_kZ^{-1})}{-(1+G_3Z^{-1})}$$

so,

$$\det A_2'\det(D_2'-C_2'A_2'^{-1}B_2') = 2K_2Z^{-2}(1+c_kZ^{-1})-(1+d_kZ^{-1})(1+G_3Z^{-1})- g_{31}(2K_2)(1+c_kZ^{-1})$$

3.0 Symbolic Gain Evaluation of Hybrid Systems

In the previous chapters, we have addressed the symbolic gain evaluation of analog and digital systems separately. But in real life, it is the combination of both analog and digital systems most abundant. The manipulation of this hybrid systems are more complicate and special methodologies are needed. For example, Sedlar and Bekey[11] proposed a direct method which allows the evaluation of input-output relations directly from the hybrid signal flow graph. Kuo[12] also presented a modified Mason's formulation. The method we proposed here use the Grassmann algebra for the symbolic gain evaluation and suitable for automated implementations in digital computers.

3.1 Fundamental Difficulties

Consider the hybrid system shown in Figure 13, there are two samplers in the position marked with y_1 and y_2 respectively. To evaluate the symbolic gain of this kind of hybrid systems, it is found that the evaluation will be more simple by using y_1 and y_2 as the state variables of the flow graph. Reconfigurate the flow graph shown in Figure 13 to get the one in Figure 14,

Then,

$$y_2 = G_1 \, [\, R+G_3 \, y_1^* \,] \tag{36}$$

and

$$y_1 = G_2 \, y_2 \tag{37}$$

where $y_1=y_1(s)$, $y_2=y_2(s)$. And y_1^*, y_2^* are the sampled version of the state variables y_1 and y_2 respectively.

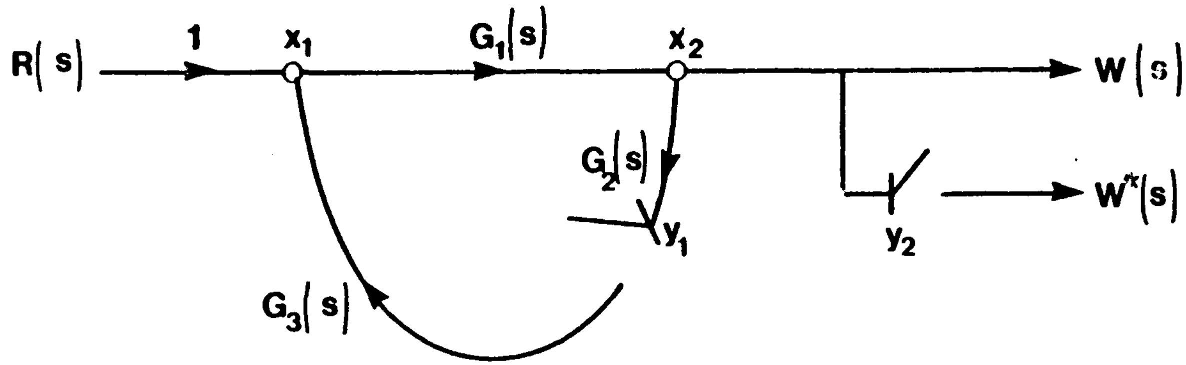


FIGURE 13

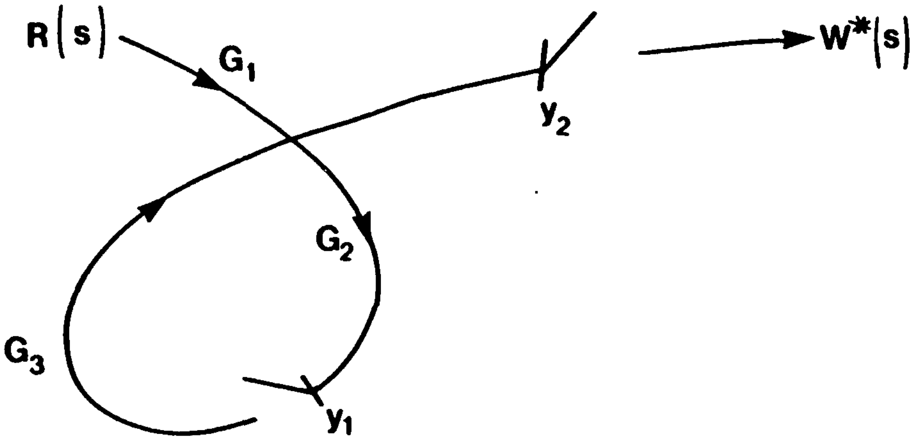


FIGURE 14

Therefore,

$$y_1/G_2 = G_1 [R + G_3 y_1^*]$$

which is equivalent to

$$y_1 = G_1G_2 [R + G_3 y_1^*] \tag{38}$$

Taking samples at both side, (38) becomes

$$y_1^* = (G_1G_2R)^* + (G_1G_2G_3)^*y_1^*$$

where $(G_1G_2R)^* = [G_1(s)G_2(s)R(s)]^*$ and $(G_1G_2R)^* = [G_1(s)G_2(s)R(S)]^*$
Note also that $(G_1G_2R)^* \neq G_1^*(s)G_2^*(s)R^*(s)$, in general.

Therefore

$$y_1^* = \frac{(G_1G_2R)^*}{1 - (G_1G_2G_3)^*} \tag{39}$$

substituting (39) into (36), we have

$$y_2 = G_1[R + G_3 \frac{(G_1G_2R)^*}{1 - (G_1G_2G^3)^*}]$$

From the flow graph shown in Figure 14,

$$W^* = y_2^*$$

then

$$W^* = (G_1R)^* + (G_1G_3)^* \frac{(G_1G_2R)^*}{1 - (G_1G_2G_3)^*} \tag{40}$$

Some literature[12] states that (40) is the pulse transfer function of the hybrid system. But, in general,

$$(G_1R)^* \neq G_1{}^*R^*$$

and $$(G_1G_2R)^* \neq (G_1G_2)^*R^*.$$

which implies that (39) is inseparable into the form

$$W^* = (T^*)R^* \tag{41}$$

The inseparability of (40) implies that the flow graph shown in Figure 14 cannot be replaced by an equivalent Z-transfer function. It makes the usefulness of (40) very limited.

To alleviate this situation, we can add an input sampler to the flow graph, it is shown in *Figure 15(a). Then, the state equations* becomes

$$y_2 = G_1 \; [\; R^* + G_3 \; y_1{}^* \;] \tag{42}$$

and $$y_1 = G_2 \; y_2 \tag{43}$$

proceed as mentioned, we have

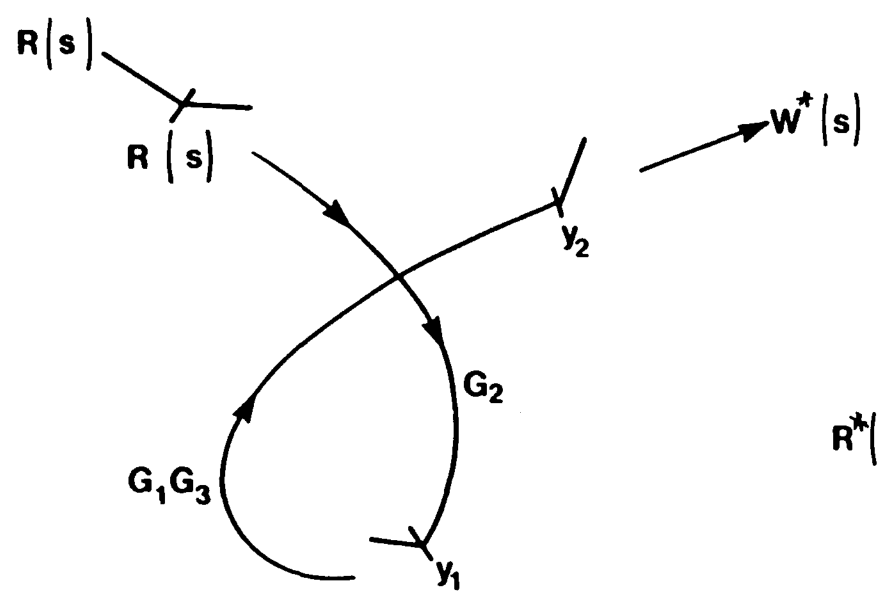
$$W^* = G_1{}^*R^* + (G_1G_3)^* \frac{(G_1G_2)^*R^*}{1 - (G_1G_2G_3)^*}$$

$$= [G_1{}^* + (G_1G_3)^* \frac{(G_1G_2)^*}{1 - (G_1G_2G_3)^*} \;]\; R^*$$

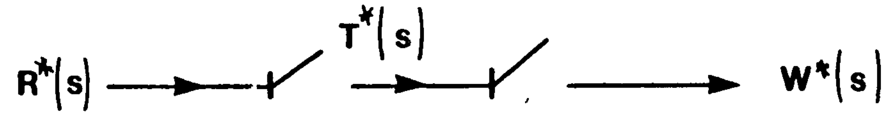
$$= T^*(s) \; R^* \tag{44}$$

using (44), the simplified flow graph with

$$T^*(s) = G_1{}^* + \frac{(G_1G_2)^*(G_1G_3)^*}{1 - (G_1G_2G_3)^*}$$



A.



B.

FIGURE 15

is obtained and shown in Figure 15(b). To get the Z-transfer function of the system, use the following substitution

$$T(Z) = T^*(s)|_{s=(\ln Z)/T}$$

Hereafter, for the reason of implementations, we assume that there are always samplers at the inputs of the hybrid systems such that the Z-transfer function of the system can be easily obtained.

3.2 Symbolic Evaluation Algorithm

Consider the example in section 3.1, the state equations (42) and (43)

$$y_2 = G_1 [R^* + G_3 y_1^*]$$

and $$y_1 = G_2 y_2$$

can be written as the following matrices form

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 0 & G_2 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ G_1G_2 & 0 \end{bmatrix} \begin{bmatrix} y_1^* \\ y_2^* \end{bmatrix} + \begin{bmatrix} 0 \\ G_1 \end{bmatrix} R^*$$

$$w^* = [\ 0\ 1\ ] \begin{bmatrix} y_1^* \\ y_2^* \end{bmatrix}$$

For a hybrid system with n samplers, if we consider the input of the i^{th} sampler as $y_i(s)$, the i^{th} state variable, and the output of the sampler as $y_i^*(s)$, the generalized form of the above equation is

$$y(s) = A_1(s)y(s) + A_2(s)y^*(s) + b(s)R^*(s) \tag{45}$$

$$w^*(s) = c^{*T}y(s) \tag{46}$$

where $y(s) = [y_1(s),\ y_2(s),\ \ldots,\ y_n(s)]^T$;

$y^*(s) = [y_1{}^*(s),\ y_2{}^*(s),\ \ldots,\ y_n{}^*(s)]^T$;

$A_1(s)$ and $A_2(s)$ are $n \times n$ matrices and $b(s)$ is $n \times 1$;

$R^*(s)$ and $w^*(s)$ are the sampled input and output;

$c^{*T}(s) = [c_1{}^*(s),\ c_2{}^*(s),\ \ldots,\ c_n{}^*(s)]$.

(45) is equivalent to

$$[I-A_1(s)]y(s) = A_2(s)\ y^*(s) + b(s)R(s)$$

or

$$\begin{aligned} y(s) &= [I-A_1(s)]^{-1}\ A_2(s)\ y^*(s) + [I-A_1(s)]^{-1}\ b(s)\ R(s) \\ &= G_{11}(s)\ y^*(s) + G_{12}(s)\ R^*(s) \end{aligned} \tag{47}$$

where $G_{11}(s) = [I-A_1(s)]^{-1}\ A_2(s)$;

$G_{12}(s) = [I-A_1(s)]^{-1}\ b(s)$.

Sampling the both sides of (47), we have

$$y^*(s) = G_{11}{}^*(s)\ y^*(s) + G_{12}{}^*(s)\ R^*(s)$$

which is equivalent to

$$[I-G_{11}{}^*(s)]y^*(s) = G_{12}{}^*(s)\ R^*(s)$$

or $y^*(s) = [I-G_{11}{}^*(s)]^{-1}\ G_{12}{}^*(s)\ R^*(s)$

Therefore, from (46)

$$w^*(s) = c^{*T}(s)\ [I-G_{11}{}^*(s)]^{-1}\ G_{12}{}^*(s)\ R^*(s) \tag{48}$$

Since

$$F(Z) = F^*(s)|_{s=(\ln Z)/T},$$

substituting $s=(\ln Z)/T$ into (48), we have

$$w^*(Z) = c^{*T}(Z)\ [I-G_{11}{}^*(Z)]^{-1}\ G_{12}{}^*(Z)\ R^*(Z) \tag{49}$$

From (49), the Z-transfer function

$$G(Z) = c^{*T}(Z)\ [I-G_{11}{}^*(Z)]^{-1}\ G_{12}{}^*(Z) \tag{50}$$

Then, using (12), (13) and (14) again, we have

$$1 + G(Z) = \frac{\det[\ I-G_{11}(Z)+G_{12}(Z)c^T(Z)\ ]}{\det[\ I-G_{11}(Z)\ ]} \tag{51}$$

Go back to the example, we have

$$G_{11}(s) = [I-A_1(s)]^{-1}A_2(s) = \begin{bmatrix} G_1G_2G_3 & 0 \\ G_1G_3 & 0 \end{bmatrix}$$

Therefore,

$$G_{11}{}^*(s) = \begin{bmatrix} (G_1G_2G_3)^* & 0 \\ (G_1G_3)^* & 0 \end{bmatrix}$$

$$G_{12}(s) = [\, I - A_1(s)\,]^{-1} b(s) = \begin{bmatrix} G_1G_2 \\ G_1 \end{bmatrix}$$

Therefore,

$$G_{12}^*(s) = \begin{bmatrix} (G_1G_2)^* \\ (G_1)^* \end{bmatrix}$$

To evaluate the gain symbolically, use (51) with

$$c^T(Z) = [\, 0,\ 1\,],$$

$$b(Z) = G_{12}(Z) = \begin{bmatrix} G_1G_2(Z) \\ G_1(Z) \end{bmatrix}$$

then

$$\begin{aligned} \Delta_d &= \det[\, I - G_{11}(Z)\,] \\ &= \det \begin{bmatrix} 1 - G_1G_2G_3(Z) & 0 \\ -G_1G_2(Z) & 1 \end{bmatrix} \\ &= 1 - G_1G_2G_3(Z) \end{aligned}$$

$$\begin{aligned} \Delta_{11} &= \det[\, I - G_{11}(Z) + bc^T\,] \\ &= \det \begin{bmatrix} 1 - G_1G_2G_3(Z) & G_1G_2(Z) \\ -G_1G_3(Z) & 1 + G_1(Z) \end{bmatrix} - \Delta_d \\ &= [\, 1 - G_1G_2G_3(Z)\,]G_1(Z) + G_1G_2(Z)G_1G_3(Z) \end{aligned}$$

Therefore,

$$\begin{aligned} G(Z) &= \Delta_{11}/\Delta_d \\ &= G_1(Z) + \frac{G_1G_2(Z)G_1G_3(Z)}{1 - G_1G_2G_3(Z)} \end{aligned}$$

We can summarized the algorithm as:

1. For each sampler in the signal flow graph, assign a state variable; the input of the sampler is a continuous variable $y_i(s)$, while the output of the sampler is the discrete variable $y_i{}^*(s)$. Notice that we assume that there is always an input sampler and we need not assign a state variable to the sampler.

2. Reconfigurate the signal flow graph such that there are only ingoing and outgoing branches from the state variables defined in (1).

3. the state equation of the resultant flow graph is of the form

$$\begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ \cdot \\ y_n \end{bmatrix} = \begin{bmatrix} A_{11}{}^1(s) & A_{12}{}^1(s) & \dots & A_{1n}{}^1(s) \\ A_{21}{}^1(s) & A_{22}{}^1(s) & \dots & A_{2n}{}^1(s) \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ A_{n1}{}^1(s) & A_{n2}{}^1(s) & \dots & A_{nn}{}^1(s) \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ \cdot \\ y_n \end{bmatrix} +$$

$$\begin{bmatrix} A_{11}{}^2(s) & A_{12}{}^2(s) & \dots & A_{1n}{}^2(s) \\ A_{21}{}^2(s) & A_{22}{}^2(s) & \dots & A_{2n}{}^2(s) \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ A_{n1}{}^2(s) & A_{n2}{}^2(s) & \dots & A_{nn}{}^2(s) \end{bmatrix} \begin{bmatrix} y^*{}_1 \\ y^*{}_2 \\ \cdot \\ \cdot \\ \cdot \\ y^*{}_n \end{bmatrix} + \begin{bmatrix} b_1(s) \\ b_2(s) \\ \cdot \\ \cdot \\ \cdot \\ b_n(s) \end{bmatrix} r^*(s)$$

$$w(s) = [\ c_1(s),\ c_2(s),\ \dots,\ c_n(s)\ ] \begin{bmatrix} y_1{}^* \\ y_2{}^* \\ \cdot \\ \cdot \\ \cdot \\ y_n{}^* \end{bmatrix}$$

If the ij^{th} elements of the matrices $A^1(s)$ and $A^2(s)$ be $A_{ij}{}^1(s)$ and $A_{ij}{}^2(s)$, then $A_{ij}{}^1(s)$ is the weight of the branch going from the input of the j^{th} sampler to the input of the i^{th} sampler. $A_{ij}{}^2(s)$ is the weight of the branch

going from the output of the j^{th} sampler to the input of the i^{th} sampler. And, $b_i(s)$ is the weight of the branch going from the output of the input sampler to the i^{th} state variable node. $c_i(s)$ is the weight of the branch going from the the output of the i^{th} sampler to the output node $w(s)$

4. Let $G_{11}(s) = [\ I-A^1(s)\ ]^{-1}\ A^2(s)$,

and $G_{12}(s) = [\ I-A^1(s)\ ]^{-1}\ b(s)$.

then the output

$$w^*(s) = c^{*T}(s)\ [\ I-G_{11}{}^*(s)\ ]^{-1}\ G_{12}{}^*(s)\ r^*(s)$$

and the Z-transfer function

$$G(Z) = c^T(Z)\ [\ I-G_{11}(Z)\ ]^{-1}\ G_{12}(Z).$$

5. To evaluate the Z-transfer function symbolically, we can use the Grassmann algebra defined in chapter 1 and the identity (51)

$$1 + G(Z) = \frac{\det[\ I-G_{11}(Z)+G_{12}(Z)c^T(Z)\ ]}{\det[\ I-G_{11}(Z)\ ]}$$

4.0 Decoupling of Multivariable Systems

In this chapter, we will explore the problem of decoupling a MIMO system. We will also concentrate in systems described by square transfer matrices because any system described by non-square transfer matrices can be decomposed into several systems described by square matrices. The main reason to the discussion of decoupling stems from our concept toward MIMO systems design. We want to decouple the MIMO system first so as to obtain an one-to one correspondence among the inputs and the outputs. Once the relation is established, the well developed SISO design methodologies can then be applied to the systems.

4.1 P-Constrained and V-Constrained System Model

In the previous Chapters, we use Grassmann algebra to obtain the following transfer matrix,

$$Y(s) = G(s)\, U(s) \tag{52}$$

If

$$G(s) = \begin{bmatrix} g_{11}(s) & g_{12}(s) & \cdots & g_{1n}(s) \\ g_{21}(s) & g_{22}(s) & \cdots & g_{2n}(s) \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ g_{n1}(s) & g_{n2}(s) & \cdots & g_{nn}(s) \end{bmatrix}$$

then the system is equivalent to the block diagram shown in Figure 16. According to Mesarovic[13], the structure shown in Figure 16 is known as a P-constrained system. For the simplicity of further explanation, we change the notations in (52) to that in (53)

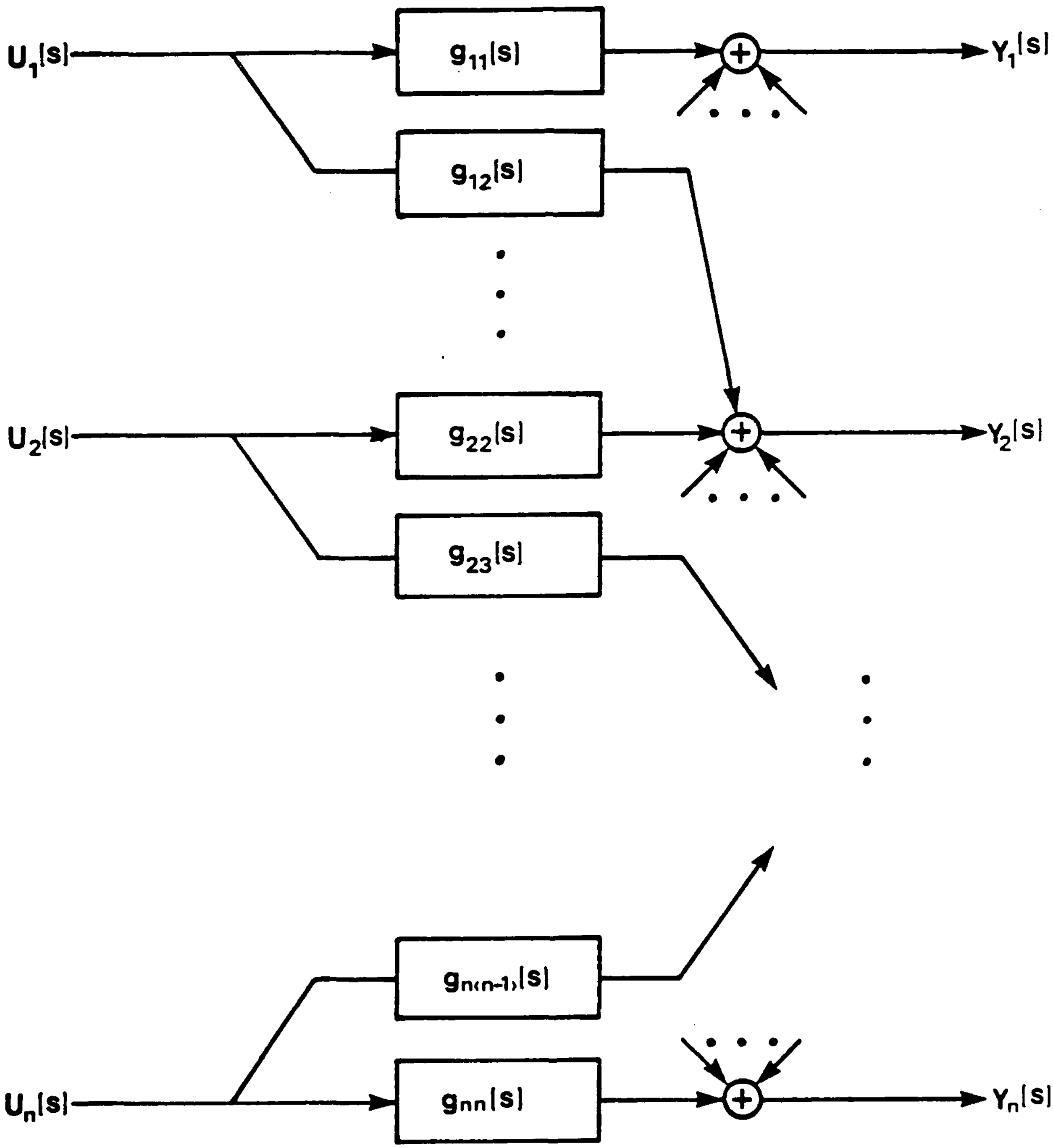


FIGURE 16

$$C_p(s) = P(s)\ M_p(s) \tag{53}$$

where $C_p(s)$ is the MIMO output matrix; $M_p(s)$ the input matrix and $P(s)$, with $P(s)=G(s)$, as the P-constrained system matrix.

There is another basic structure of MIMO systems known as V-constrained systems. The block diagram of this kind of systems is given in Figure 17(a). The main reason to investigate the P- and V- constrained system structures is that they need different decoupling strategies to obtain effectiveness and simplicities. All these will be explored in the later sections. If we let, as Figure 17(b),

$$H = \begin{bmatrix} V_{11} & 0 & 0 & \dots & 0 \\ 0 & V_{22} & 0 & \dots & 0 \\ . & 0 & V_{33} & \dots & 0 \\ . & . & . & & 0 \\ 0 & 0 & 0 & \dots & V_{nn} \end{bmatrix}, \text{ and } K = \begin{bmatrix} 0 & V_{12} & V_{13} & \dots & V_{1n} \\ V_{21} & 0 & V_{23} & \dots & V_{2n} \\ V_{31} & V_{32} & 0 & \dots & V_{3n} \\ . & . & . & & . \\ V_{n1} & V_{n2} & . & \dots & 0 \end{bmatrix}$$

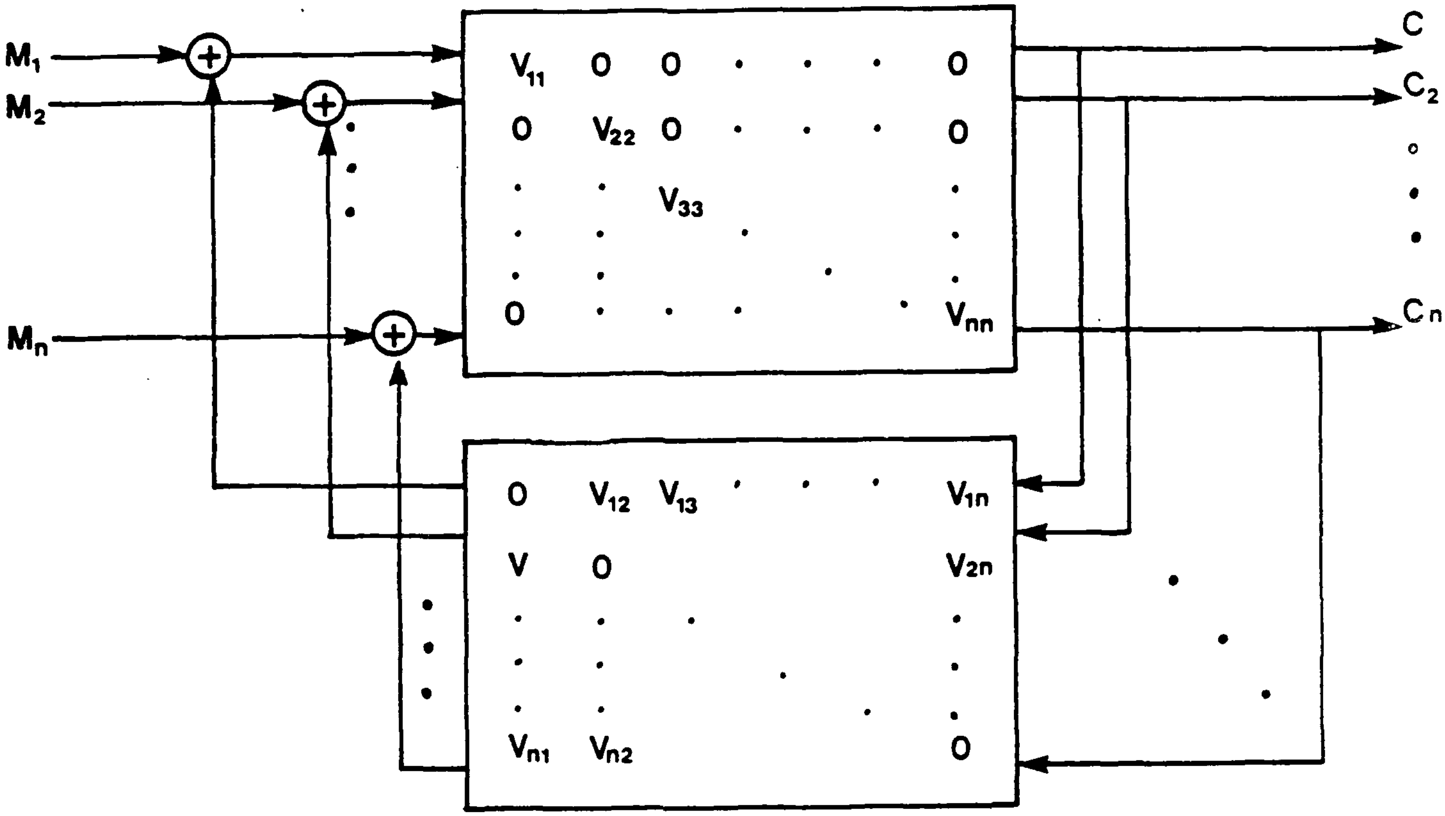
then, we have

$$C_V = H\ M_V + H\ K\ C_V \tag{54}$$

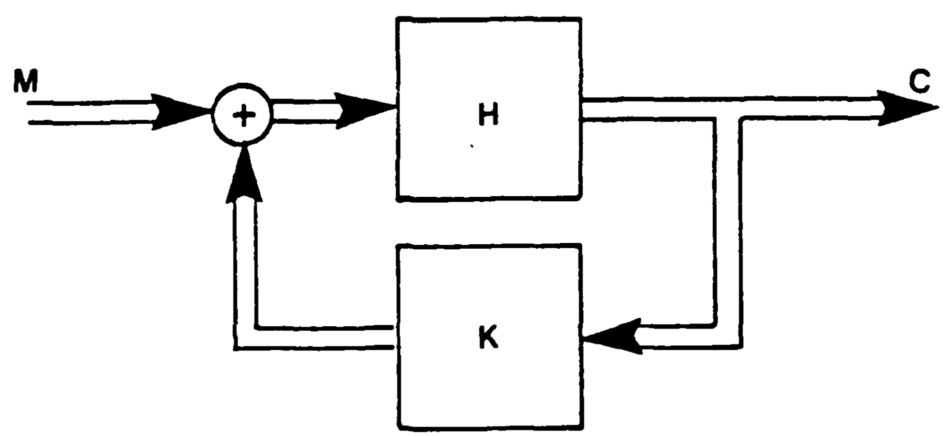
or $$C_V = [\ I - H\ K\ ]^{-1}\ H\ M_V \tag{55}$$

Comparing (53) and (55), if both systems are subjected to the same input signals, i.e, $M_p = M_V$, then the condition to obtain identical responses will be

$$P = [\ I- H\ K\ ]^{-1}\ H \tag{56}$$


M1
M2
Mn
V11 0 0 0
0 V22 0 0
V33
0 Vnn
0 V12 V13 V1n
V 0 V2n
Vn1 Vn2 0
C
C2
Cn

A.


M
H
K
C

B.

FIGURE 17

If the matrices P and H in (56) are nonsingular, then taking the inverse of (56), we have

$$
\begin{aligned}
P^{-1} &= H^{-1} [\, I-HK \,] \\
&= H^{-1} - K
\end{aligned}
\tag{57}
$$

Since

$$P^{-1} = \frac{\text{adj } P}{\det P},$$

where adj P is the adjacent matrix of P.

Let

$$\text{adj } P = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ A_{n1} & A_{n2} & \cdots & A_{nn} \end{bmatrix}$$

then (57) implies

$$H^{-1} = \frac{1}{\det P} \begin{bmatrix} A_{11} & 0 & 0 & \cdots & 0 \\ 0 & A_{22} & 0 & \cdots & 0 \\ \cdot & \cdot & \cdot & & \cdot \\ 0 & 0 & 0 & \cdots & A_{nn} \end{bmatrix} \tag{58a}$$

$$\text{and} \quad K = \frac{-1}{\det P} \begin{bmatrix} 0 & A_{12} & A_{13} & \cdots & A_{1n} \\ A_{21} & 0 & A_{23} & \cdots & A_{2n} \\ \cdot & \cdot & \cdot & & \cdot \\ A_{n1} & A_{n2} & \cdot & \cdots & A_{nn} \end{bmatrix} \tag{58b}$$

If we denote the diagonal matrix constructed by taking the diagonal elements of any matrix, P, as $[P]_D$, then (58a) and (58b) are simplified as

$$H^{-1} = [\, P^{-1} \,]_D \tag{59a}$$

$$\text{and} \quad K = [\, P^{-1} \,]_D - P^{-1} \tag{59b}$$

From (56), if all the matrices are invertible, it is impossible to identify whether the MIMO system is P- or V- constrained. It is because for any given matrix P, there are always matrices H and K such that (56) is satisfied, or vice verse. But we are going to show, through the following two examples, that whether a system is P- or V- constrained is determined by the physical structure of the system.

Example 4

Examine Figure 18, it is the schematic of a fluid mixing control process. The same fluid with different temperature T_1 and T_2 are inputed from A and B respectively. It is hoped that the fluid with temperature T and flow Q is outputed from C. The problem is to obtained a preseted T_o and Q_o by controlling the flows Q_1 and Q_2 through the valves in 1 and 2.

The flow of the fluid is preserved, that is

$$Q = Q_1 + Q_2$$

In steady state, let

$$Q_o = Q_{1o} + Q_{2o}$$

where Q_{1o} and Q_{2o} are the steady state valves of Q_1 and Q_2 respectively.

The steady state error

$$e_1 = \Delta Q/Q_o = \frac{Q_{1o}}{Q_o}\,\frac{\Delta Q_1}{Q_{1o}} + \frac{Q_{2o}}{Q_o}\,\frac{\Delta Q_2}{Q_{2o}} \qquad (60)$$

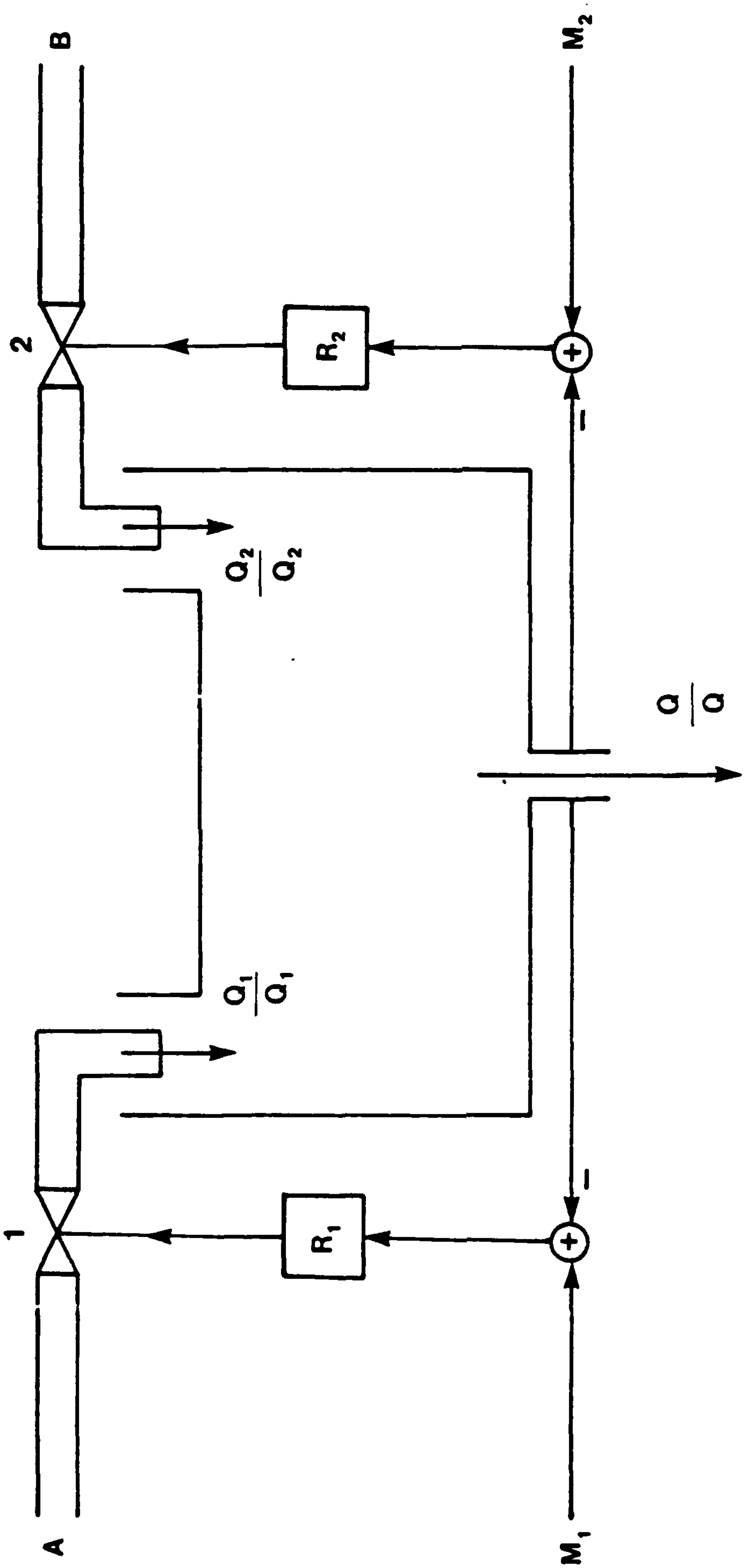


FIGURE 18

where ΔQ, ΔQ_1 and ΔQ_2 are the deviations from Q_o, Q_{1o}, and Q_{2o} respectively.

$$K_1 = \frac{Q_{1o}}{Q_o},\quad K_2 = \frac{\Delta Q_1}{Q_{1o}},\quad q_1 = \frac{Q_{2o}}{Q_o},\quad q_2 = \frac{\Delta Q_2}{Q_{2o}}.$$

Since $K_1 + K_2 = 1$,

then

$$e_1 = K_1\, q_1 + (1-K_1)\, q_2 \tag{61}$$

where q_1 and q_2 are controlled by regulators R_1 and R_2 respectively.

From thermo equilibrium considerations, we have

$$(Q_1 + Q_2)T = Q_1T_1 + Q_2T_2$$

Since $\qquad T = T(Q_1, Q_2)$

$$\Delta T = (T - T_o)$$

$$= \frac{\partial T}{\partial Q_1}\Delta Q_1 + \frac{\partial T}{\partial Q_2}\Delta Q_2$$

$$= \frac{Q_{1o}Q_{2o}}{Q_o^2}(T_2 - T_1)\left[\frac{\Delta Q_2}{Q_{2o}} - \frac{\Delta Q_1}{Q_{1o}}\right]$$

$$= T_m\left[\frac{\Delta Q_2}{Q_{2o}} - \frac{\Delta Q_1}{Q_{1o}}\right]$$

where $T_m = \dfrac{Q_{1o}Q_{2o}}{Q_o^2}(T_2 - T_1)$.

Then the steady state temperature error is

$$e_2 = \frac{\Delta T}{T_m} = \frac{\Delta Q_2}{Q_{2o}} - \frac{\Delta Q_1}{Q_{1o}} = q_2 - q_1 \tag{63}$$

From (61) and (63), we obtain Figure 19 and the following matrix equation

$$\begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \begin{bmatrix} K_1 & 1-K_1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$$

which is a P-constrained system structure.

Example 5

Examine Figure 20, the objective is to control the high of the fluid, h, and the output flow, Q_o, by varying y_i and y_o; where y_i and y_o are quantities proportional to the variable cross areas of the input and output valves respectively. The input flow of the fluid is denoted by Q_i and the surface area of the container be a constant A.

We have

$$\Delta Q_o = C_{11}\, \Delta y_i + C_{12}\, \Delta h \tag{64}$$

where C_{11} and C_{12} are positive proportional constants.

Since

$$A\, dh = Q_i\, dt - Q_o\, dt$$

After a time t elapsed,

$$h = \frac{1}{A} \int_0^t (Q_i - Q_o)dt + h_o$$

where h_o is the initial high of the fluid.

That is,

$$\Delta h = h - h_o = \frac{1}{A}\int_0^t (\Delta Q_i - \Delta Q_o)dt$$

by $\quad \Delta Q_i = C_i \, \Delta y_i$

where C_i is a positive proportional constant.

We have

$$\Delta h = \frac{1}{A}\int_0^t (C_i \Delta y_i - \Delta Q_o)dt \tag{65}$$

Taking the Laplace transforms of (64) and (65), we obtain

$$\Delta Q_o(s) = C_{11} \, [\, \Delta y_o(s) + (\, C_{12}/C_{11} \,)\Delta h(s) \,]$$

$$\Delta h(s) = (C_i/As)[\, \Delta y_i(s) - (1/C_i)\Delta Q_o(s) \,]$$

which is equivalent to the following matrix equation

$$\begin{bmatrix} \Delta Q_o \\ \Delta h \end{bmatrix} = \begin{bmatrix} C_{11} & 0 \\ 0 & (C_i/As) \end{bmatrix} \begin{bmatrix} \Delta y_o \\ \Delta y_i \end{bmatrix} + \begin{bmatrix} C_{11} & 0 \\ 0 & (C_i/As) \end{bmatrix} \begin{bmatrix} 0 & (C_{12}/C_{11}) \\ (-1/C_i) & 0 \end{bmatrix} \begin{bmatrix} \Delta Q_o \\ \Delta Q_o \end{bmatrix}$$

Comparing with (54), we have

$$H = \begin{bmatrix} C_{11} & 0 \\ 0 & (C_i/As) \end{bmatrix}, \text{ and } \quad K = \begin{bmatrix} 0 & (C_{12}/C_{11}) \\ (-1/C_i) & 0 \end{bmatrix}$$

That is, it has a V-constrained system structure. The block diagram of this system is shown in Figure 21.

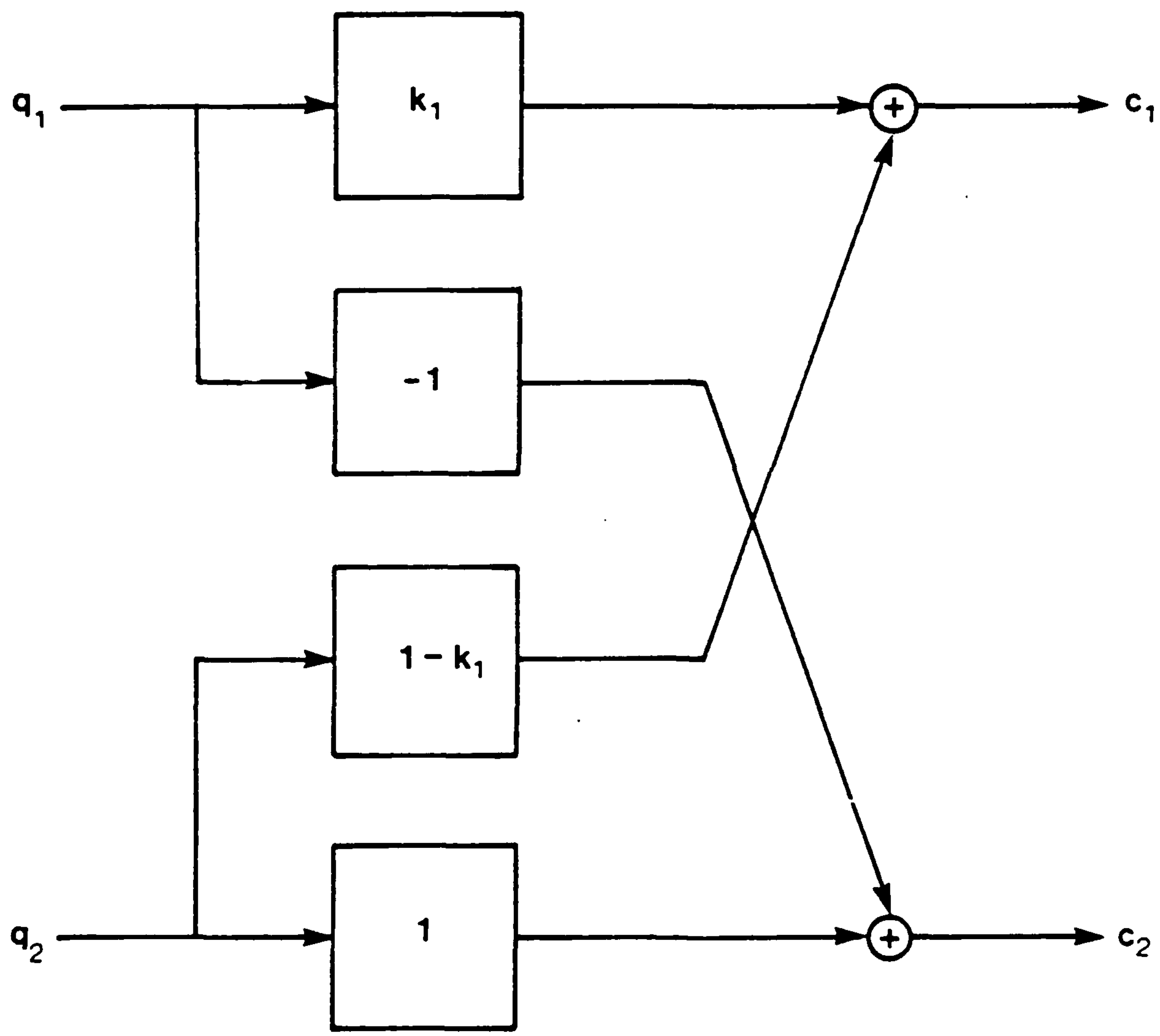


FIGURE 19

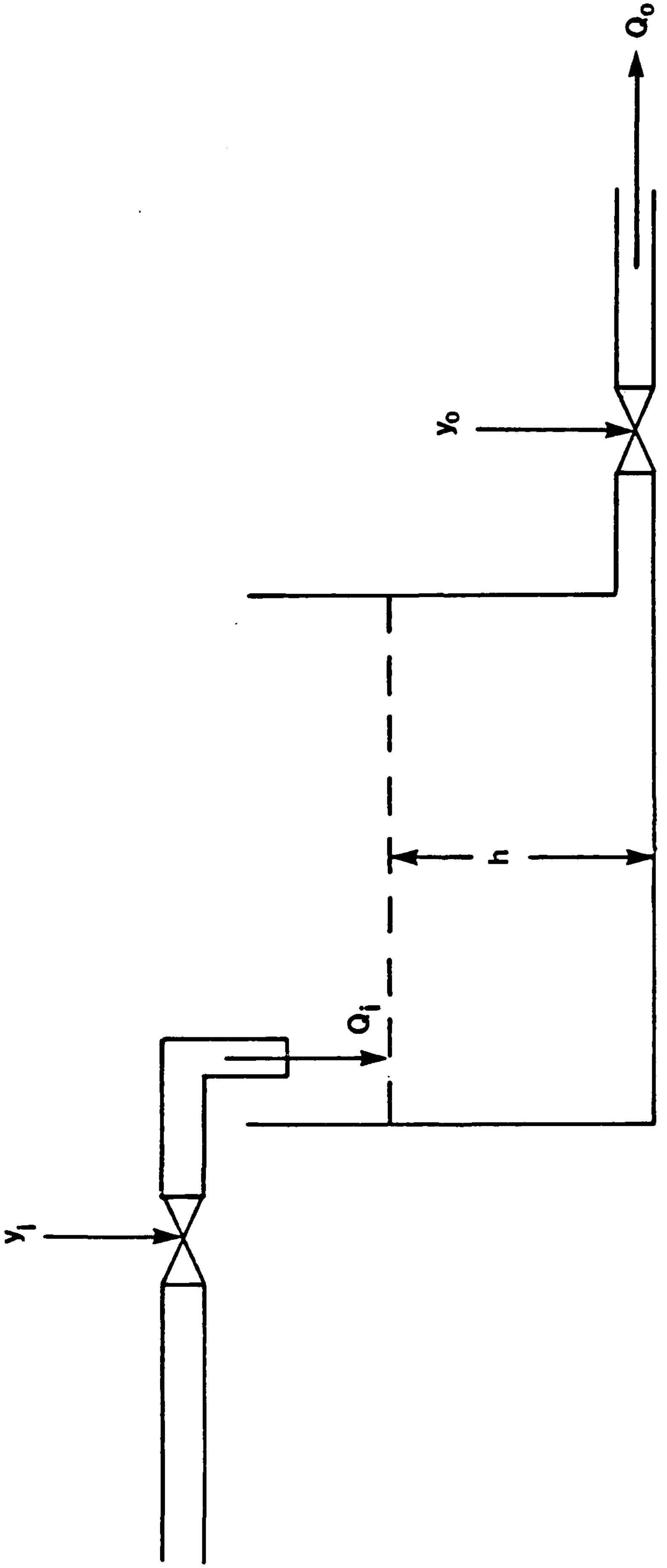


FIGURE 20

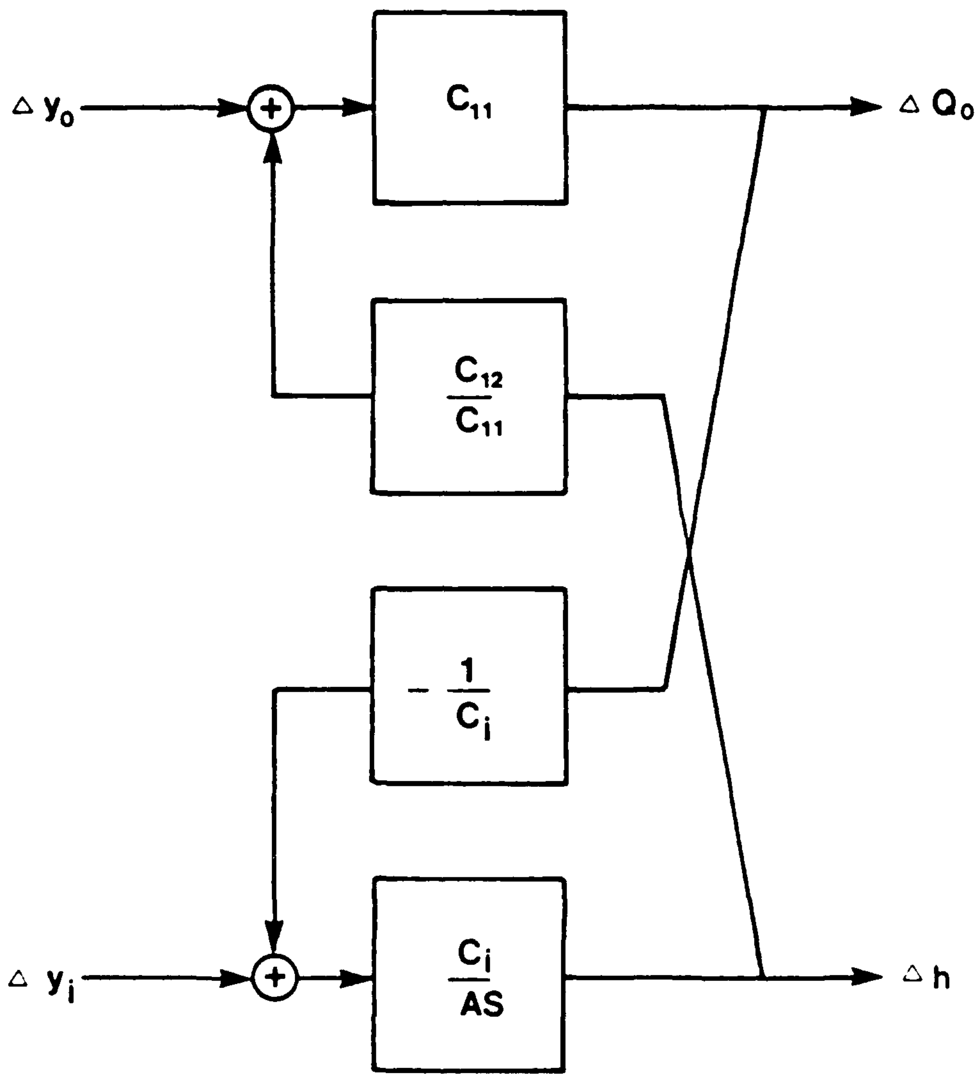


FIGURE 21

4.2 Symbolic Conversions

When we need to convert a P-constrained mathematical model to a V-constrained one, we can use the Grassmann algebra so that symbolic evaluation is accomplished. By examinating (59a) and (59b), once we obtain the P^{-1}, the H^{-1} and K are easily constructed. Therefore, the symbolic evaluation becomes a symbolic matrix inverse procedure.

Example 8

Consider C = P M,

where

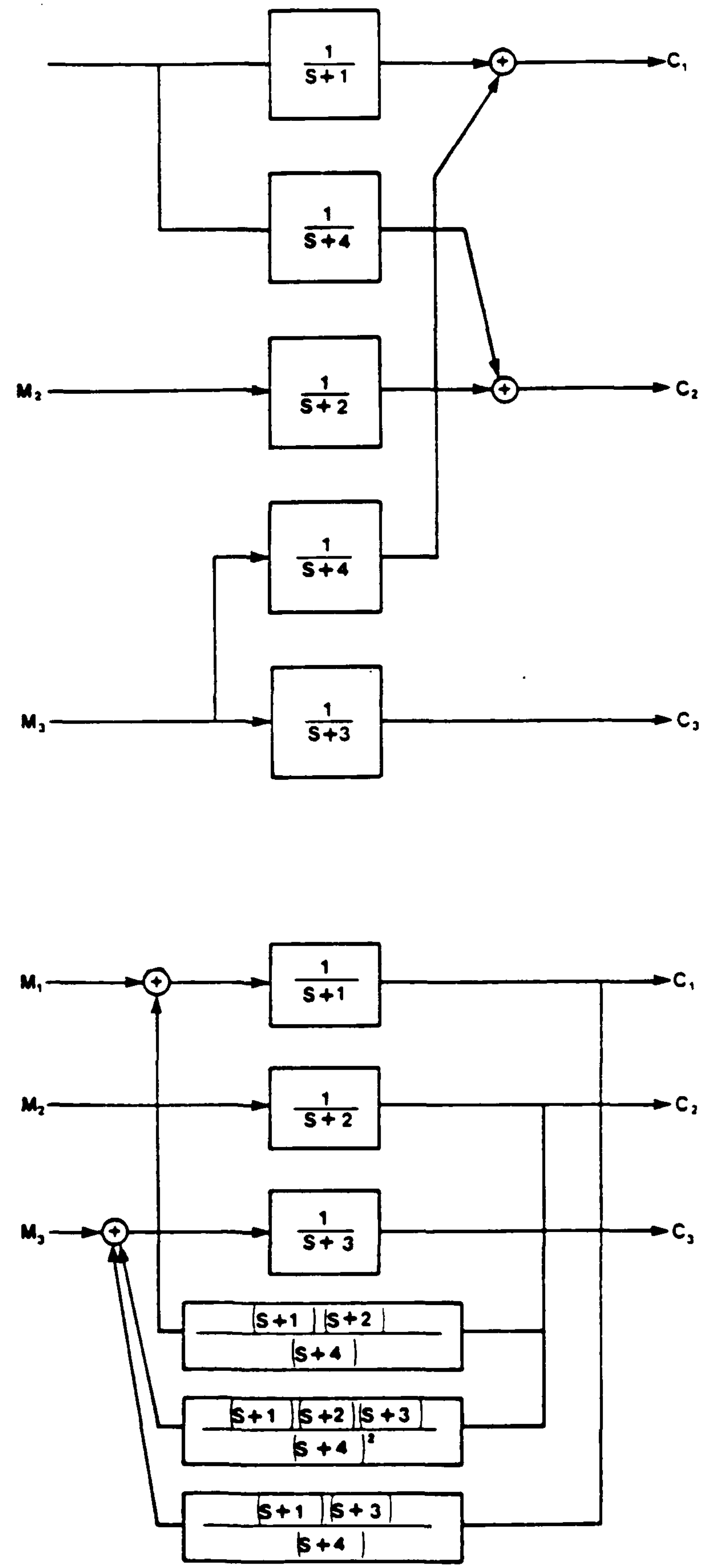
$$P = \begin{bmatrix} \dfrac{1}{(s+1)} & \dfrac{1}{(s+4)} & 0 \\ 0 & \dfrac{1}{(s+2)} & 0 \\ \dfrac{1}{(s+4)} & 0 & \dfrac{1}{(s+3)} \end{bmatrix}$$

The block diagram of this P-constrained structure is shown in Figure 22(a). To obtain the H and K used in the V-constrained description (54), we need to symbolically evaluate the inverse of P.

Using Grassmann algebra with $C^T = B = I$, that is

$$c_1^T = [1\ 0\ 0],\quad c_2^T = [0\ 1\ 0],\quad c_3^T = [0\ 0\ 1],$$

and $\quad b_1 = [1\ 0\ 0]^T,\ b_2 = [0\ 1\ 0]^T,\ b_3 = [0\ 0\ 1]^T.$


1/(S+1)
1/(S+4)
M2
1/(S+2)
1/(S+4)
M3
1/(S+3)
C1
C2
C3
M1
M2
M3
1/(S+1)
1/(S+2)
1/(S+3)
(S+1)(S+2)/(S+4)
(S+1)(S+2)(S+3)/(S+4)^2
(S+1)(S+3)/(S+4)
C1
C2
C3

A.

B.

FIGURE 22

then

$$\Delta_d = \det P = \frac{1}{(s+1)(s+2)(s+3)}$$

and

$$P^{-1} = \begin{bmatrix} (s+1) & \dfrac{-(s+1)(s+2)}{(s+4)} & 0 \\ 0 & (s+2) & 0 \\ \dfrac{-(s+1)(s+3)}{(s+4)} & \dfrac{(s+1)(s+2)(s+3)}{(s+4)^2} & (s+3) \end{bmatrix}$$

then from (59a) and (59b), we have

$$H^{-1} = \begin{bmatrix} (s+1) & 0 & 0 \\ 0 & (s+2) & 0 \\ 0 & 0 & (s+3) \end{bmatrix}$$

$$K = \begin{bmatrix} 0 & \dfrac{(s+1)(s+2)}{(s+4)} & 0 \\ 0 & 0 & 0 \\ \dfrac{(s+1)(s+3)}{(s+4)} & \dfrac{-(s+1)(s+2)(s+3)}{(s+4)^2} & 0 \end{bmatrix}$$

The resultant block diagram of the equivalent V-constrained system is given in Figure 22(b).

To convert from the V-constrained structure into the P-constrained one, we can, of course, use (56). But it is simpler to start from (57). It is illustrated by the following example.

Example 7

Consider the following V-constrained structure with

$$H = \begin{bmatrix} \frac{1}{(s+2)} & 0 & 0 \\ 0 & \frac{1}{(s+4)} & 0 \\ 0 & 0 & \frac{1}{(s+6)} \end{bmatrix}$$

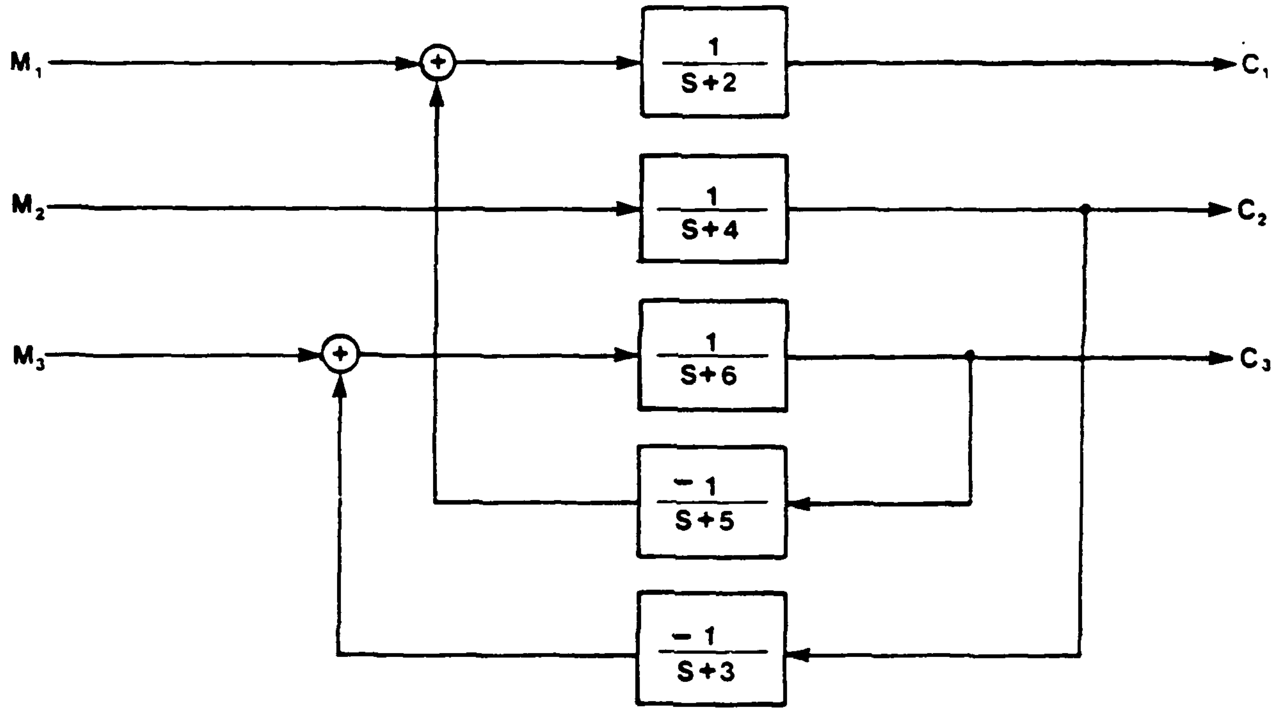
and

$$K = \begin{bmatrix} 0 & 0 & \frac{-1}{(s+5)} \\ 0 & 0 & 0 \\ 0 & \frac{-1}{(s+3)} & 0 \end{bmatrix}$$

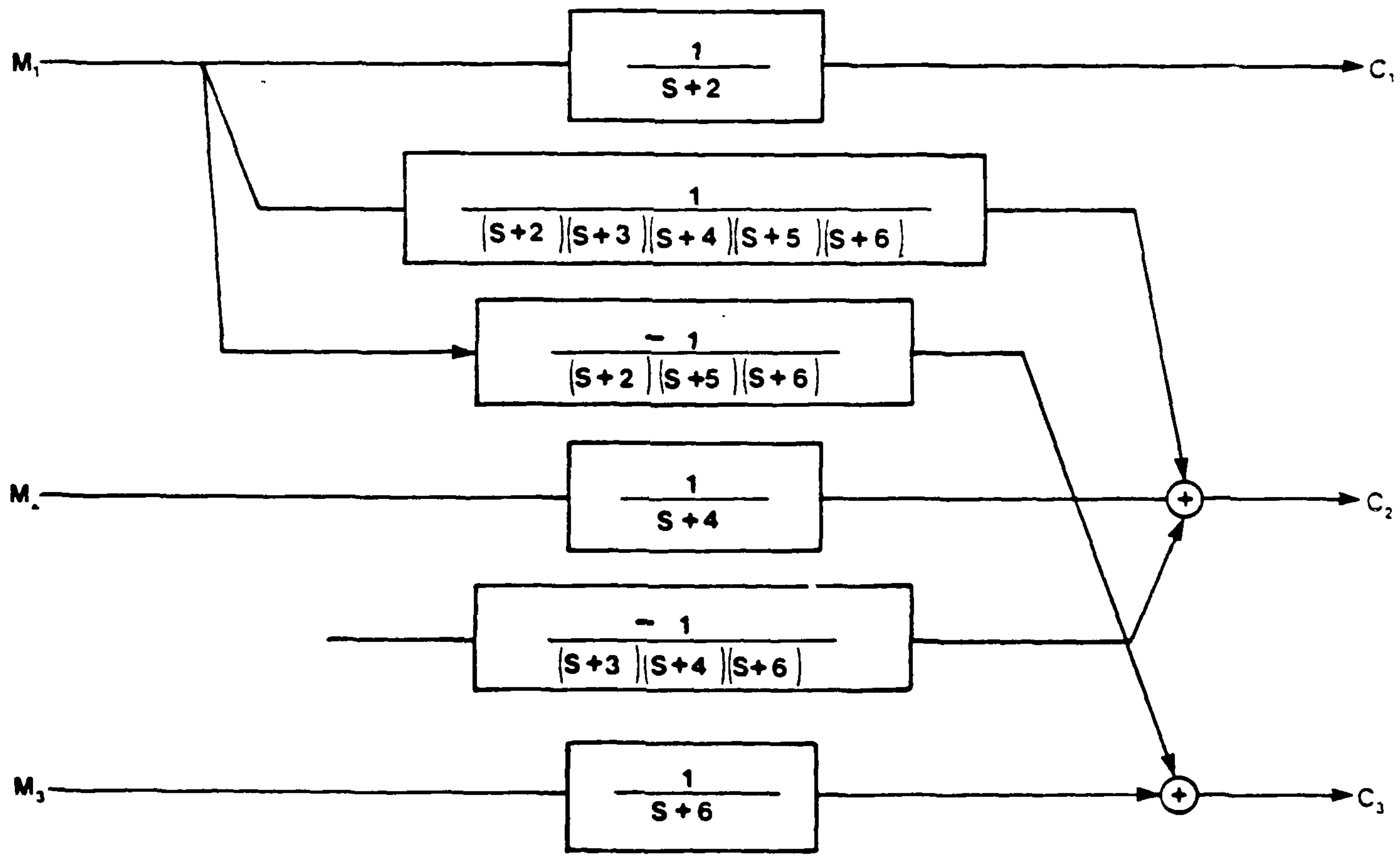
The block diagram of this system is given in Figure 23(a). By (57)

$$P^{-1} = H^{-1} - K = \begin{bmatrix} (s+2) & 0 & \frac{1}{(s+3)} \\ 0 & (s+4) & 0 \\ 0 & \frac{1}{(s+3)} & (s+6) \end{bmatrix}$$

Once again, we can use the Grassmann algebra to obtain the P symbolically. Proceeds as the previous example, we have


M_1
M_2
M_3
C_1
C_2
C_3
$\frac{1}{S+2}$
$\frac{1}{S+4}$
$\frac{1}{S+6}$
$\frac{-1}{S+5}$
$\frac{-1}{S+3}$

A.


M_1
M_2
M_3
C_1
C_2
C_3
$\frac{1}{S+2}$
$\frac{1}{(S+2)(S+3)(S+4)(S+5)(S+6)}$
$\frac{-1}{(S+2)(S+5)(S+6)}$
$\frac{1}{S+4}$
$\frac{-1}{(S+3)(S+4)(S+6)}$
$\frac{1}{S+6}$

B.

Fig. 23

$$P = \begin{bmatrix} \frac{1}{(s+2)} & \frac{1}{(s+2)(s+3)(s+4)(s+5)(s+6)} & \frac{-1}{(s+2)(s+5)(s+4)} \\ 0 & \frac{1}{(s+4)} & 0 \\ 0 & \frac{-1}{(s+3)(s+4)(s+6)} & \frac{1}{(s+6)} \end{bmatrix}$$

The resultant P-constrained structure is shown in figure 23(b).

4.3 P-Constrained and V-Constrained Decoupling Structures

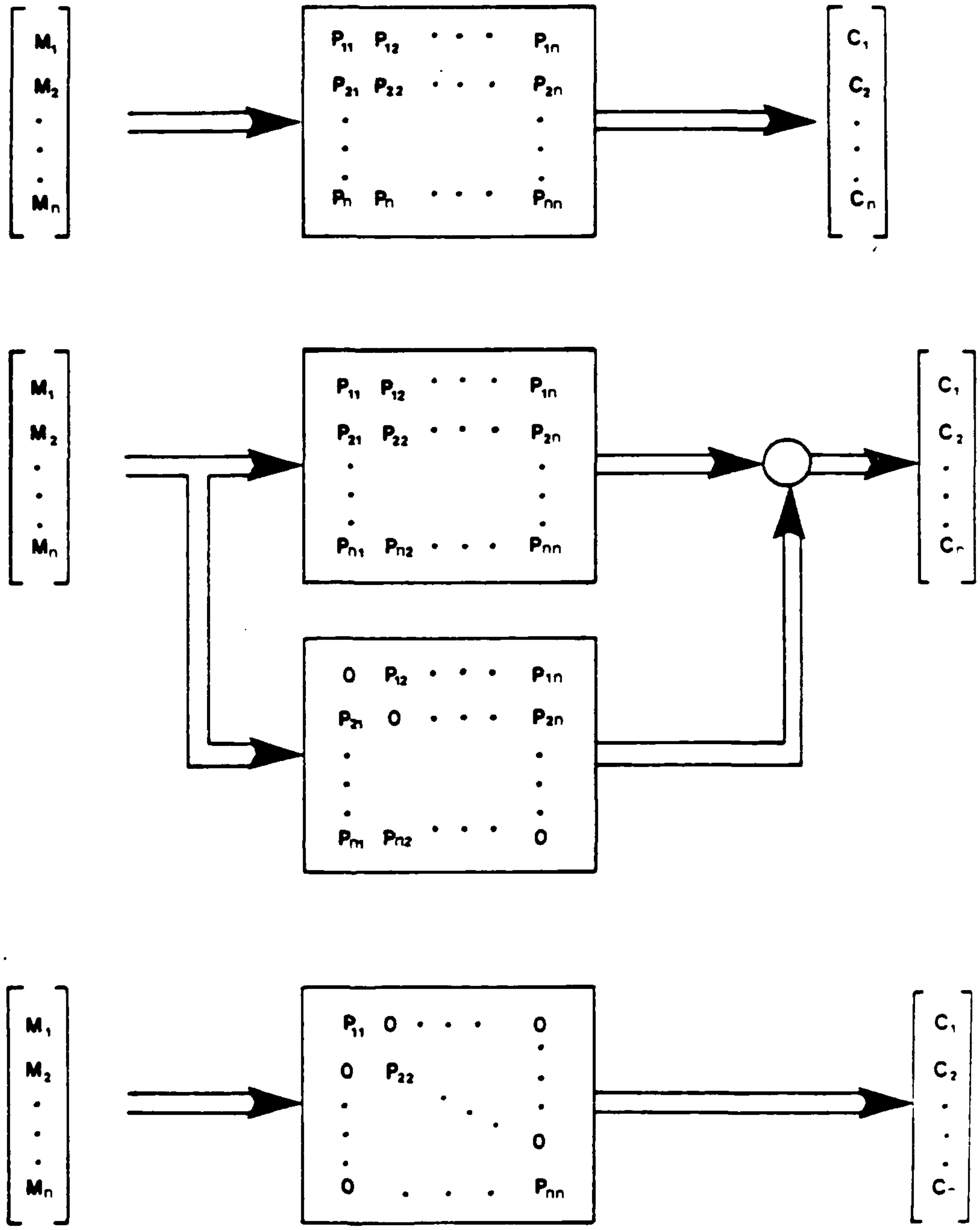
Consider (53) again

$$Cp(s) = P(s)\ Mp(s) \qquad (53)$$

When the elements $p_{ij}(s)=0$ for all $i \neq j$, that is, P(s) is a diagonal matrix, we say that the system (53) is decoupled. For an open loop system, decoupling can be easily achieved. For example, examine Figure 24(a), it is a P-constrained system. If we add a feedforward path in this open loop structure, as shown in Figure 24(b), the system will be decoupled and is shown in Figure 24(c).

For an open loop V-constrained system, as the one given in Figure 25(a), we can add a feedback path to cancel out the K matrix as given in (54). The feedback path is shown in Figure 25(b) and the decoupled system is shown in Figure 25(c).

But, in practice, we use a lots of closed loop systems. The presentation of a simple closed-loop system decoupling method is the objective of this section.



M1
M2
Mn
P11 P12 P1n
P21 P22 P2n
Pnn
C1
C2
Cn
0 P12 P1n
P21 0 P2n
0
P11 0 0
0 P22
0 Pnn

A.

B.

FIGURE 24

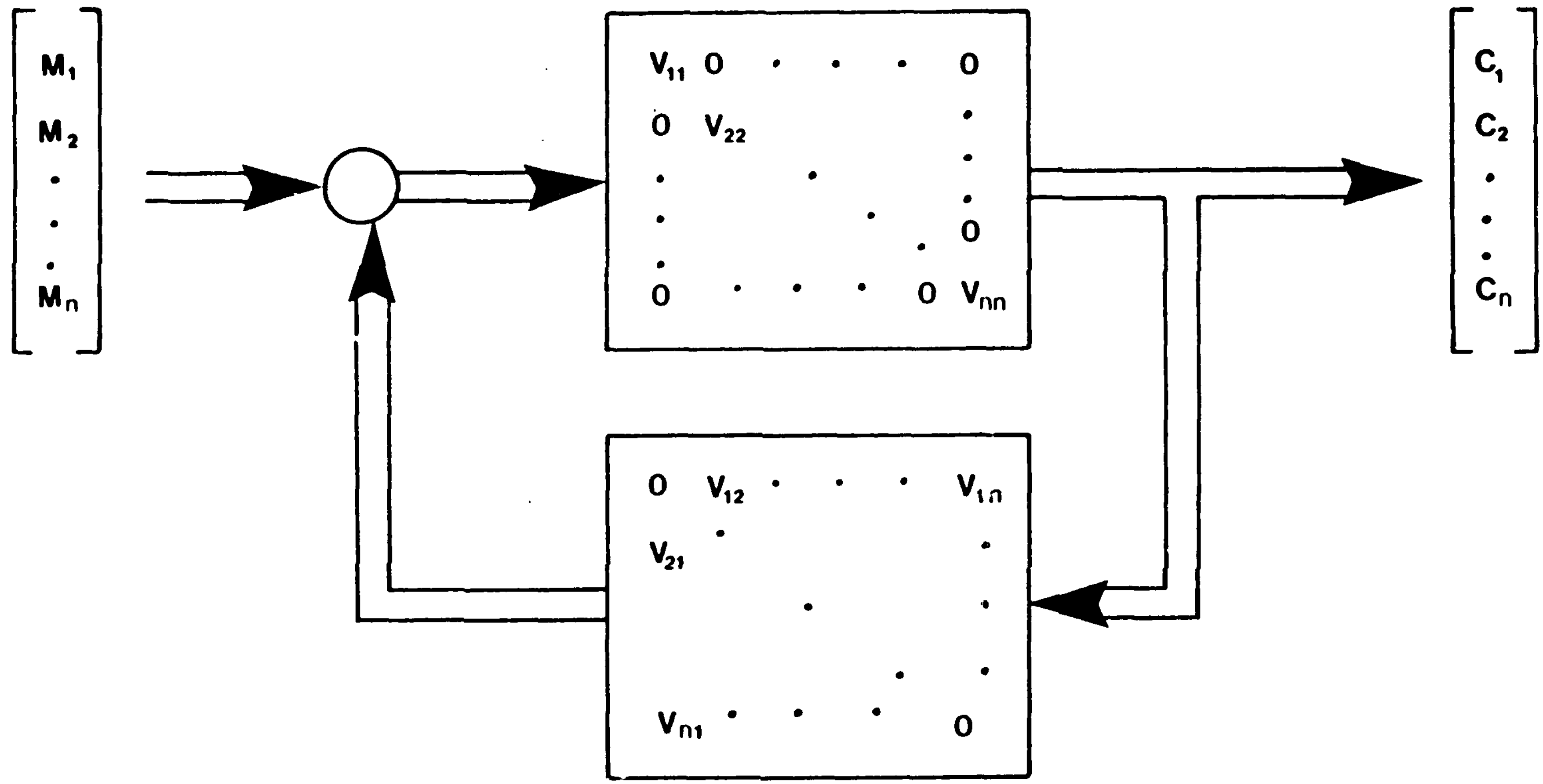


FIGURE 25 A.

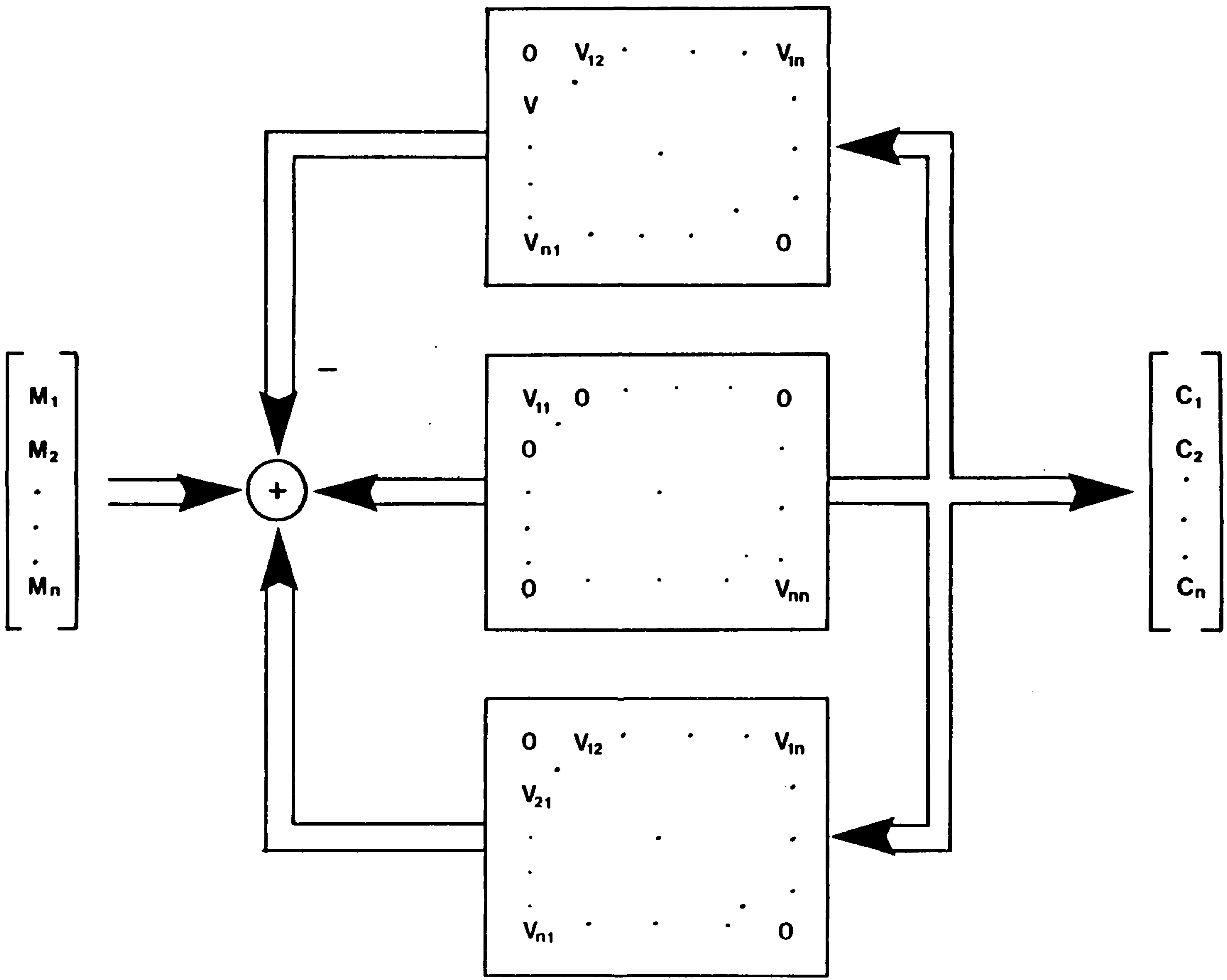


FIGURE 25B.

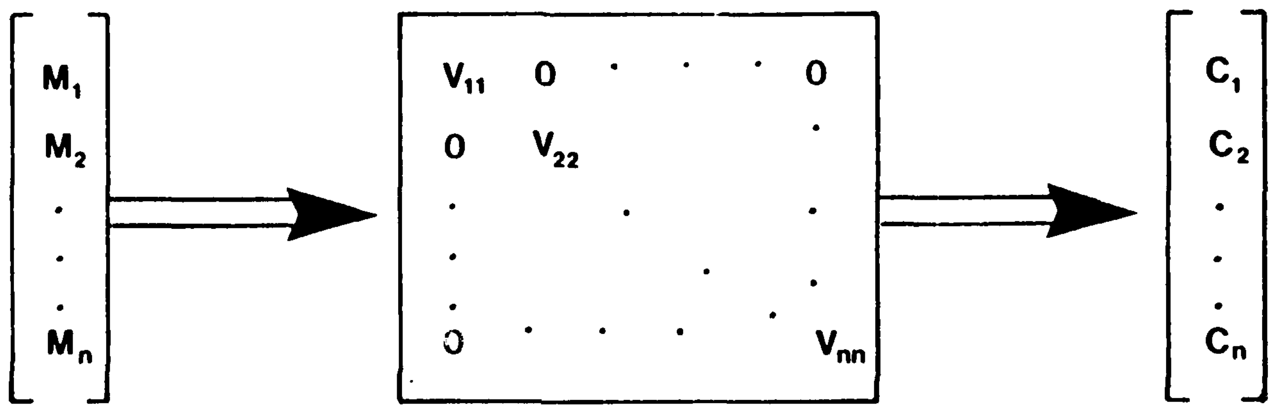


FIGURE 25 C.

To decouple the closed loop system shown in Figure 26(a), we can put a decoupling structure N between the compensator R and the plant P, as shown in Figure 26(b). Notice that the matrices R and F are unknown yet, but we can impose the condition that R and F are diagonal. It is because once the loop is decoupled, we need only to determine the corresponding diagonal elements in R and F by SISO design methods.

Since the couplings of MIMO systems can be divided into P- or V- constrained structures, as illustrated in the last sections, we can use the same structures in the decoupler N. That is, we assume that there are P-constrained and V-constrained decoupler structures. These structures of decouplers are illustrated in Figure 27(a) and (b).

Now, by the assumption that R and F are diagonal, we want to obtain the condition that N will decoupled the closed loop system shown in 26(a). From the figure, we have

$$C = PNR(M - FC)$$

which is equivalent to

$$[\ I+PNRF\ ]C = PNRM$$

or, $$C = [\ I+PNRF\ ]^{-1}\ PNRM$$
$$= G\ M$$

where G is the transfer matrix of the closed loop system.

That is,

$$G = [\ I+PNRF\ ]^{-1}\ PNR \tag{66}$$

We want the closed loop system decoupled, which is equivalent to require that G is diagonal. Since R and F are diagonal, we want to prove that G is diagonal if and only if the matrix PN is diagonal.

Firstly, we want to prove that if PN is diagonal then G is diagonal.

since the addition and multiplication of diagonal matrices are also diagonal, the matrices $W = I+PNRF$ and $Z = PNR$ are diagonal. Then, (66) becomes

$$G = W^{-1}\ Z$$

Since

$$W^{-1} = \begin{bmatrix} w_{11} & 0 & 0 \dots & 0 \\ 0 & w_{22} & 0 \dots & 0 \\ . & . & & . \\ . & . & & . \\ 0 & 0 & \dots & w_{nn} \end{bmatrix}^{-1} = \begin{bmatrix} 1/w_{11} & 0 & 0 \dots 0 \\ 0 & 1/w_{22} & 0 \dots 0 \\ . & . & . \\ . & . & . \\ 0 & 0 & \dots\ 1/w_{nn} \end{bmatrix}$$

That is W^{-1} is also diagonal. G is equal to the multiplication of two diagonal matrices and therefore diagonal.

Now, we want to prove that if G is diagonal then PN is diagonal.

(66) is equivalent to

$$[\ I+PNRF\ ]G = PNR$$

or $\quad PNR - PNRFG = G$

or $\quad PN = [\ R-RFG\ ]^{-1}G \qquad (67)$

Since G is diagonal, the right hand side of (67) are all diagonal matrices. Follow the arguments as the former prove, we must have PN is diagonal. When the plant has a V-constrained structure, we can use (56) or $P=[I-HK]^{-1}H$ in all the proofs and the same results hold.

To summarize(refer to Figure 26a), with the matrices R and F diagonal, to decouple the closed loop system is equivalent to design a decoupler N such that the matrix PN is diagonal.

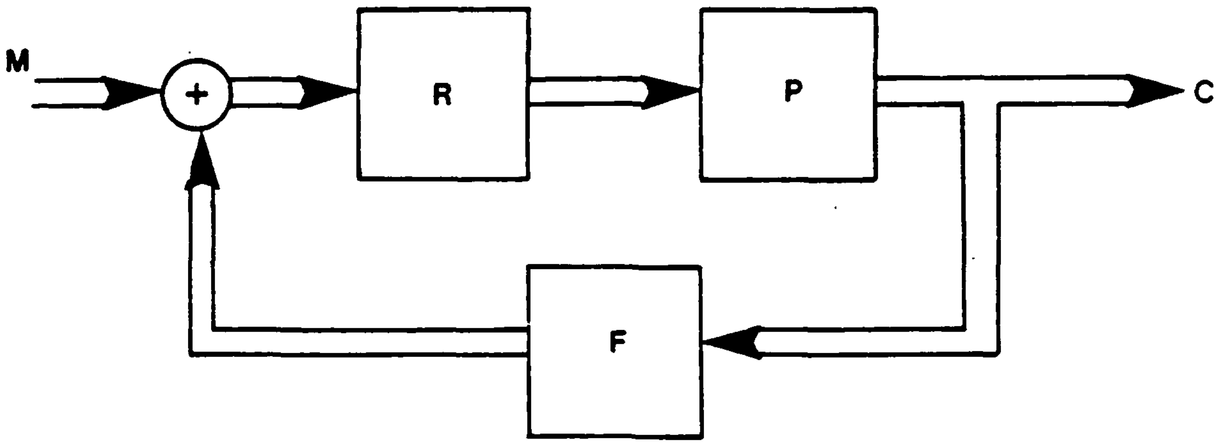
Let $[P]_D$ be the diagonal matrix of P by eliminating all the non-diagonal elements, then a solution to make PN diagonal is

$$N = P^{-1}\ [P]_D \qquad (68)$$

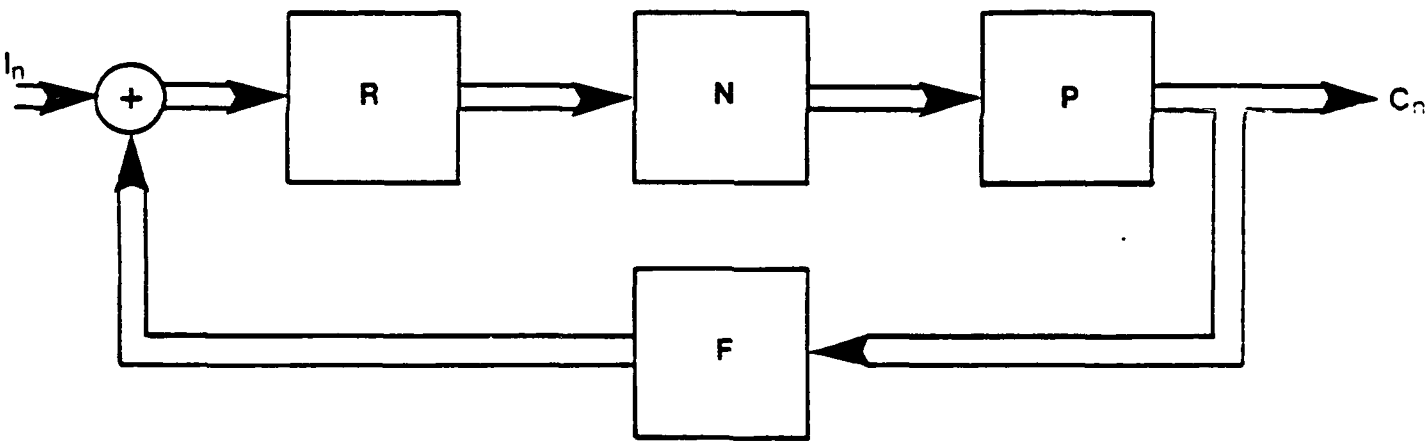
because

$$PN = P\ P^{-1}[P]_D = [P]_D.$$

We are interested in simple implementions of decouplers in the form of (68).

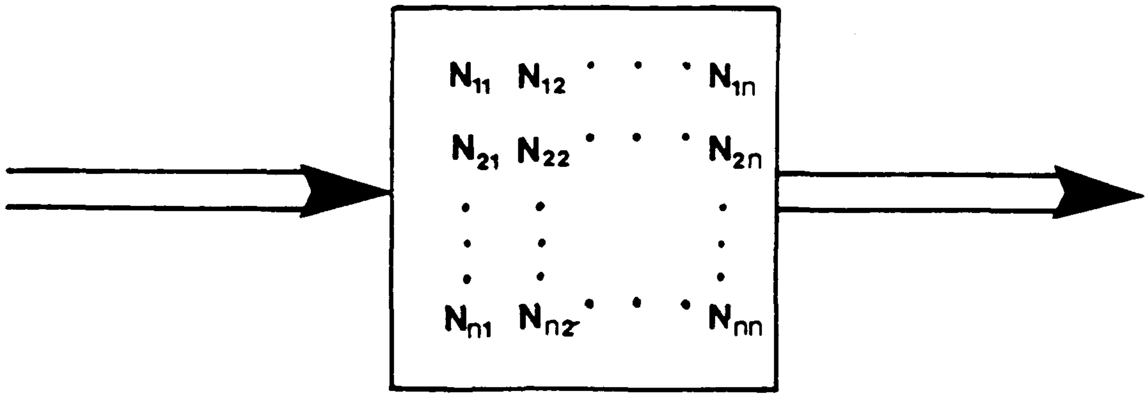


A.

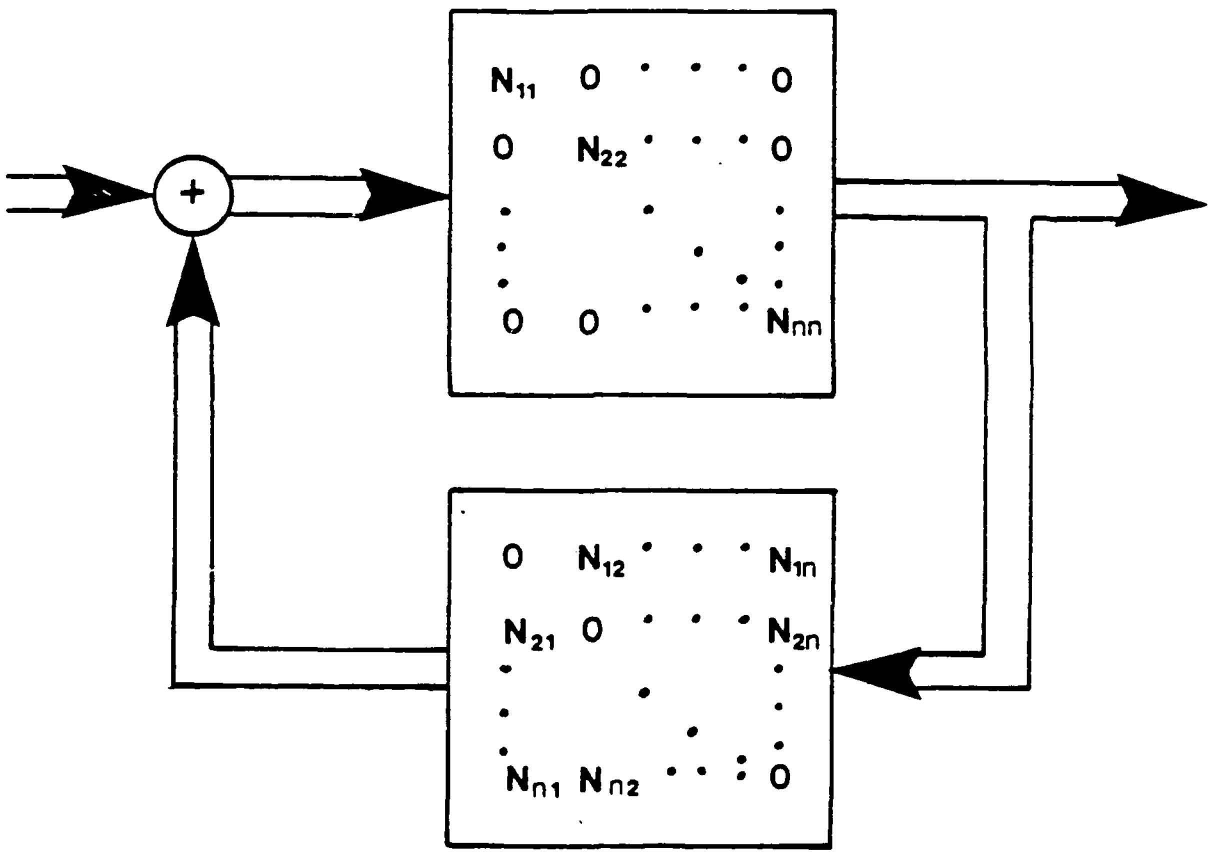


B.

FIGURE 26


N_{11} N_{12} · · · N_{1n}
N_{21} N_{22} · · · N_{2n}
N_{n1} N_{n2} · · · N_{nn}

B.


+
N_{11} 0 · · · 0
0 N_{22} · · · 0
0 0 · · · N_{nn}
0 N_{12} · · · N_{1n}
N_{21} 0 · · · N_{2n}
N_{n1} N_{n2} · · · 0

A.

FIGURE 27

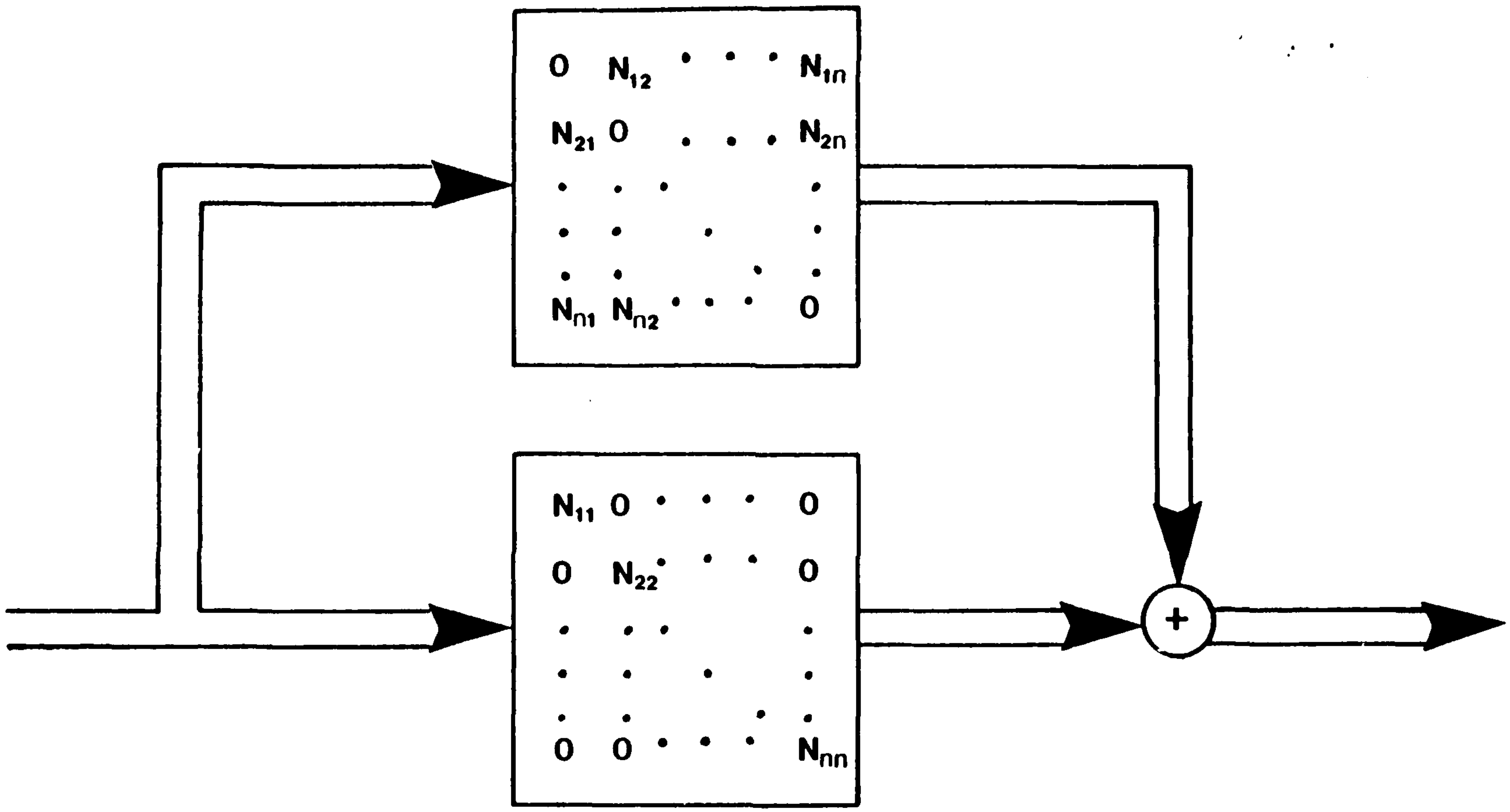
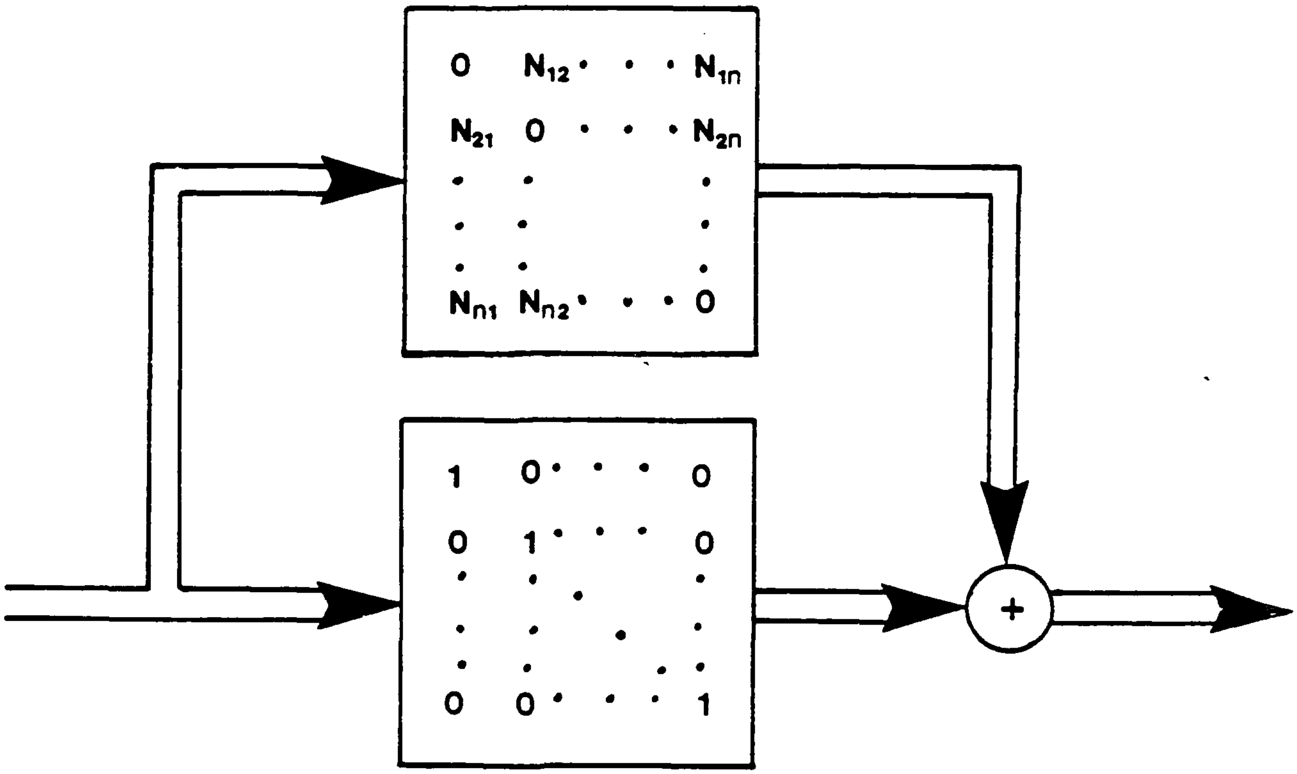
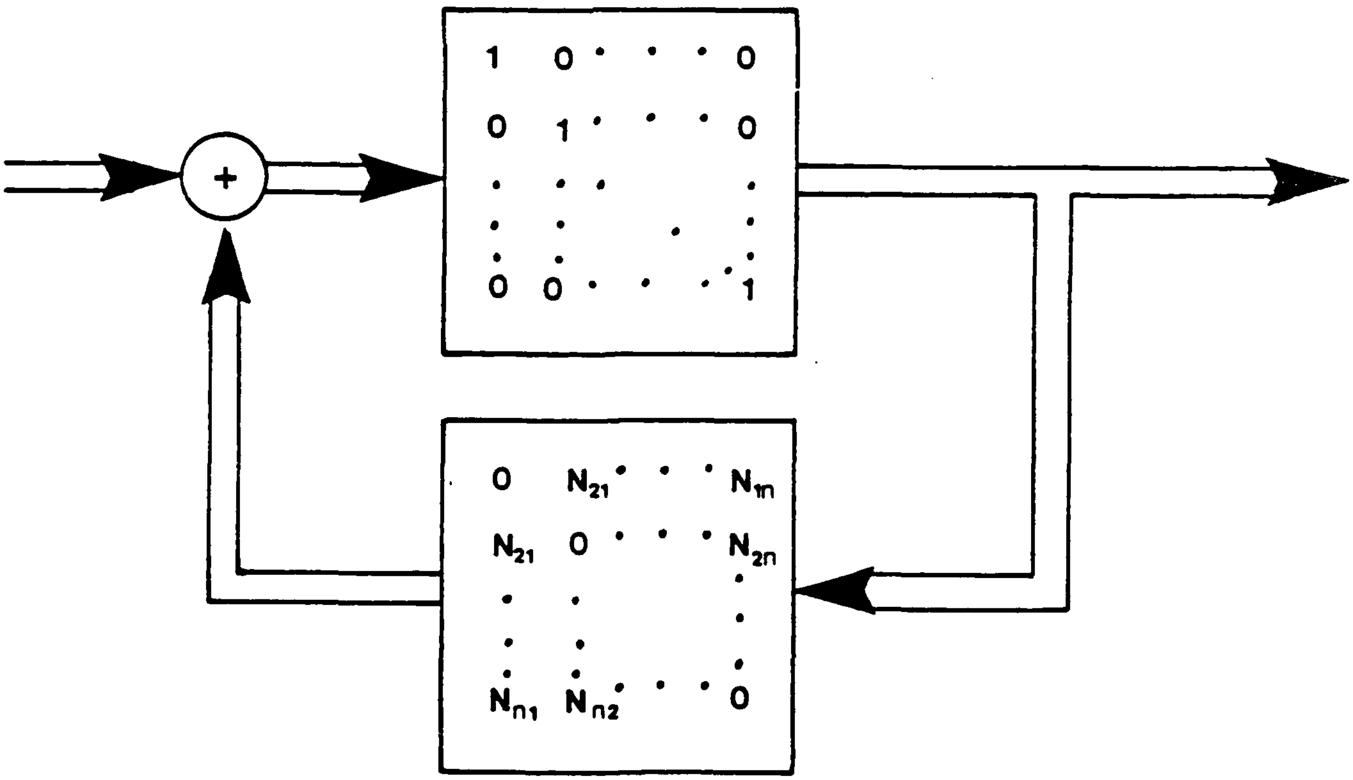


FIGURE 28


0 N12 · · · N1n
N21 0 · · · N2n
Nn1 Nn2 · · · 0
1 0 · · · 0
0 1 · · · 0
0 0 · · · 1
+

A.


+
1 0 · · · 0
0 1 · · · 0
0 0 · · · 1
0 N21 · · · N1n
N21 0 · · · N2n
Nn1 Nn2 · · · 0

B.

FIGURE 29

Let us go back to Figure 27(a) and (b). We can reconfigure Figure 27(a) to obtain Figure 28. It is important to notice that Figure 28 is a feedforward structure and Figure 27(b) is a feedback structure. For the simplicity of implementation, we are especially interested in the one shown in Figure 28 with all the diagonal elements equal to unity; and, for the structure shown in Figure 27(b), we are interested in the one with H=I. These decoupling structures are shown in Figure 29(a) and (b) respectively.

From Figure 29(a), it is easy to see that the P-constrained decoupler becomes a feedforward path from the outputs of the compensator R to the inputs of the plant P. From Figure 29(b), we also notice that the V-constrained decoupler becomes a feedback path from the inputs of the plant P to the outputs of the compensator R. These feedforward and feedback decoupling structures are simple implementations of (68). In the next section, we are going to characterize these structures and some explicitly formulae are also derived.

4.4 The Structure of Decoupled systems

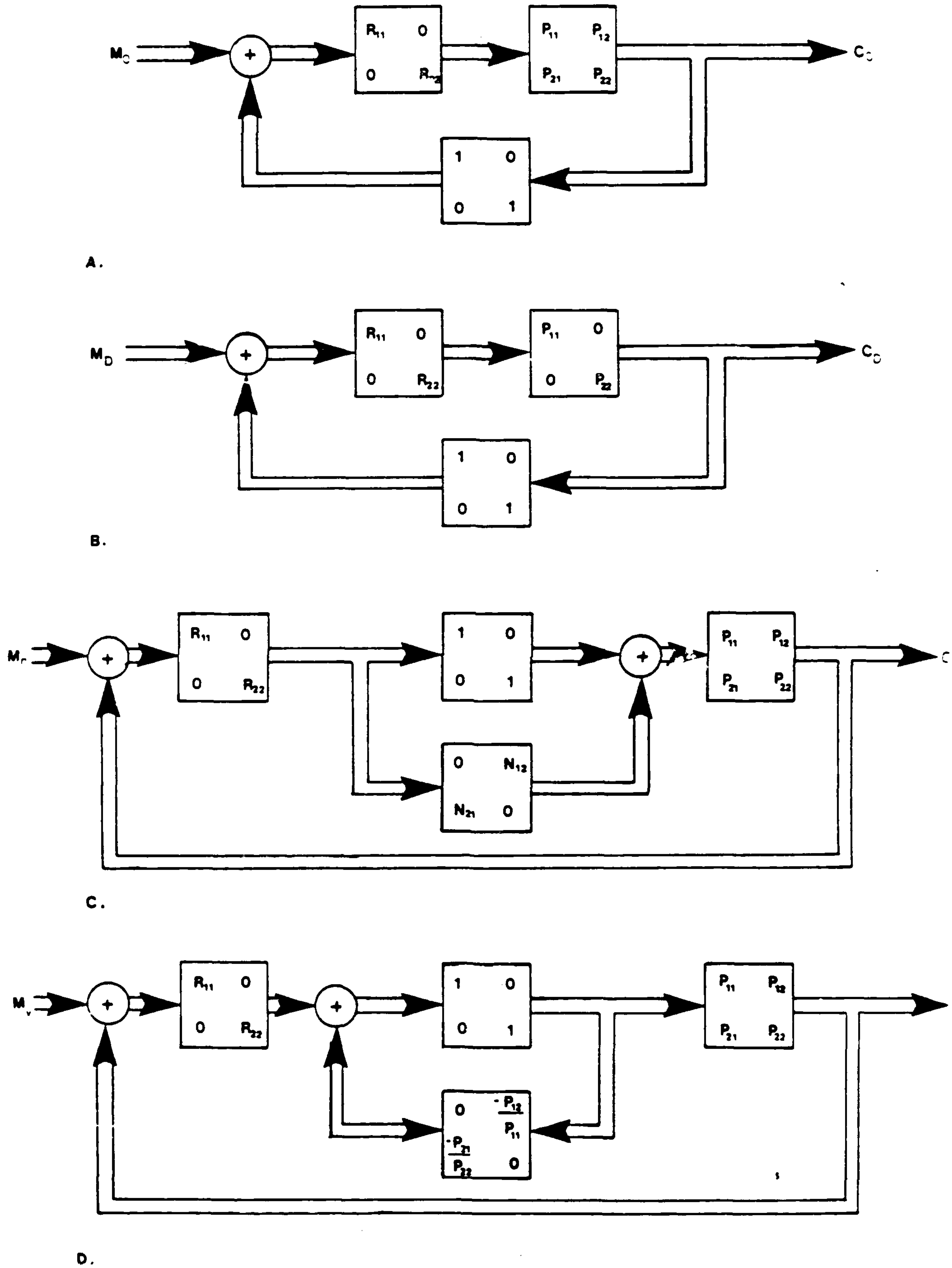
In the previous sections, we intended to decouple a MIMO system into a set of SISO systems. But what is the structures of these systems? From an implementation point of view, we would like to have the set of SISO systems after decoupling equal to the set obtained by simply opening all the coupling channels of the MIMO system. It is because we can save a lots of time in evaluating the resultant set of systems after the decoupling. The set of SISO systems obtain by opening the coupling channels is at hand well before the decoupler is designed. Indeed, we have virtually separated the process of decoupler design and the SISO systems design by this requirement. As mentioned in the last section, we are only interested in the decoupler structures given in Figure 29(a) and (b). This section will give the explicit design formulae of the decouplers and the limitations are stated.

For more understanding, consider the 2-inputs and 2-outputs closed loop system given in Figure 30(a)

Opening the coupling channels is equivalent to make P_{12} and P_{21} zero and the equivalent block diagram is shown in Figure 30(b). The transfer matrix of this opened system can be found by

$$C_d = [\, I+P_D R]^{-1} P_D R M_d$$

$$= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1+R_{11}P_{11} & 0 \\ 0 & 1+R_{22}P_{22} \end{bmatrix}^{-1} \begin{bmatrix} R_{11}P_{11} & 0 \\ 0 & R_{22}P_{22} \end{bmatrix} M_d$$



M_D
C_D
R11 0
0 R22
P11 P12
P21 P22
1 0
0 1
A.
M_D
C_D
R11 0
0 R22
P11 0
0 P22
1 0
0 1
B.
R11 0
0 R22
1 0
0 1
0 N12
N21 0
P11 P12
P21 P22
C.
M_v
R11 0
0 R22
1 0
0 1
0 -P12/P11
-P21/P22 0
P11 P12
P21 P22
D.

FIGURE 30

$$= \begin{bmatrix} \dfrac{R_{11}P_{11}}{1+R_{11}P_{11}} & 0 \\ 0 & \dfrac{R_{22}P_{22}}{1+R_{22}P_{22}} \end{bmatrix} M_d \qquad (69)$$

When a P-constrained decoupler, as shown in Figure 29(a), is inserted into Figure 30(a), Figure 30(c) is obtained. The transfer matrix is given by

$$C_n = [\ I+PNR\ ]^{-1}PNRM_n$$

Since

$$PNR = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{bmatrix} \begin{bmatrix} R_{11} & 0 \\ 0 & R_{22} \end{bmatrix}$$

$$= \begin{bmatrix} R_{11}(P_{11}N_{11} + P_{21}N_{21}) & R_{22}(P_{11}N_{12} + P_{12}N_{22}) \\ R_{11}(P_{21}N_{11} + P_{22}N_{21}) & R_{22}(P_{21}N_{12} + P_{22}N_{22}) \end{bmatrix}$$

$$= \begin{bmatrix} Z_1 & Z_2 \\ Z_3 & Z_4 \end{bmatrix}$$

Using Grassmann algebra with

$$c_1^T = [\ 1,0\ ],\ c_2^T = [\ 0,1\ ]$$

and $b_1 = [\ Z_1,Z_3\ ]^T,\ b_2 = [\ Z_2,Z_4\ ]^T$

then

$$\Delta_d = \det[\ I+PNR\ ]$$
$$= (1+Z_1)(1+Z_4) - Z_2Z_3$$

and we have

$$C_n = \begin{bmatrix} \dfrac{(1+Z_4)Z_1 - Z_2Z_3}{\Delta_d} & \dfrac{Z_2}{\Delta_d} \\[2ex] \dfrac{Z_3}{\Delta_d} & \dfrac{(1+Z_1)Z_4 - Z_2Z_3}{\Delta_d} \end{bmatrix}$$

To have the system decoupled, we need

$$Z_2 = Z_3 = 0$$

that is

$$N_{22}P_{12} + N_{12}P_{11} = 0$$

and $N_{11}P_{21} + N_{21}P_{22} = 0$

For our decoupler, $N_{11} = N_{22} = 1$, then

$$N_{12} = -P_{12}/P_{11}, \quad \text{and } N_{21} = -P_{21}/P_{22}.$$

Now the decoupled system is

$$C_n = \begin{bmatrix} \dfrac{Z_1}{1+Z_1} & 0 \\[2ex] 0 & \dfrac{Z_4}{1+Z_4} \end{bmatrix} M_n$$

$$= \begin{bmatrix} \dfrac{R_{11}(P_{11}P_{22}-P_{12}P_{21})}{P_{22}+R_{11}(P_{11}P_{22}-P_{12}P_{21})} & 0 \\[2ex] 0 & \dfrac{R_{22}(P_{11}P_{22}-P_{12}P_{21})}{P_{11}+R_{22}(P_{11}P_{22}-P_{12}P_{21})} \end{bmatrix} M_n \qquad (70)$$

When the inputs $M_d = M_n$, from (69) and (70), it is clear that $C_d \neq C_n$. That means, by using the decoupling structure in Figure 30(c), we cannot make the set of SISO systems described by (70) equals to the set of SISO systems obtained by opening all the coupling channels in Figure 30(a).

The reason that we cannot decouple the system as we want is very simple. Since the structure of the decoupler is

$$N = \begin{bmatrix} 1 & -P_{12}/P_{11} \\ -P_{21}/P_{22} & 1 \end{bmatrix}$$

then

$$PN = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} 1 & -P_{12}/P_{11} \\ -P_{21}/P_{22} & 1 \end{bmatrix}$$

$$= \begin{bmatrix} P_{11}-(P_{12}P_{21}/P_{22}) & 0 \\ 0 & P_{22}-(P_{12}P_{21}/P_{11}) \end{bmatrix}$$

which is not equal to $[P]_D$

$$[P]_D = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}$$

Now, examine Figure 30(d). We inserted a V-constrained structure decoupler before the plant matrix P. We put

$$H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \text{ and } \quad K = \begin{bmatrix} 0 & -P_{12}/P_{11} \\ -P_{21}/P_{22} & 0 \end{bmatrix}$$

then

$$PN = P[\ I-K\ ]^{-1}$$

$$= \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} 1 & P_{12}/P_{11} \\ P_{21}/P_{22} & 1 \end{bmatrix}^{-1}$$

$$= \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} 1 & -P_{12}/P_{11} \\ -P_{21}/P_{22} & 1 \end{bmatrix} (P_{11}P_{22}/(P_{11}P_{22}-P_{12}P_{21}))$$

$$= \begin{bmatrix} P_{11} & 0 \\ 0 & P_{22} \end{bmatrix}$$

which means that this closed loop system can now be decoupled into the form (69) and our requirements are achieved by using a V-constrained decoupler.

Since we have 2 decoupler structures and 2 plant structures, there will be four combinations (P- and P-, P-and V-, V-and P-, V-and V-). We will show that in order to obtain our desired decoupled systems, we need to use a V-constrained decoupler for a P-constrained plant, or vice verse.

From this example, we know that, in order to decouple a P-constrained plant structure and, at the same time, to fulfill all our requirements, all we need is to require the matrix product

$$PN = [P]_D \tag{71a}$$

but now N has only 2 allowable form. One is $N_{ii}=1$ for all i and the other is $N=[I-K]^{-1}$, where K has all diagonal elements zero. To decouple a V-constrained plant structure, we need to modify (71a) to

$$[\ I-HK\ ]^{-1}HN = H \tag{71b}$$

it has the meaning that the coupling feedback matrix K is disconnected. Furthermore, to simplify our formulation, we need a short handed notation. Given any matrix A, the matrix $[A]_D$ is the one with only the diagonal elements of A.

I. P-constrained decoupler structure vs. P-constrained plant structure.

The block diagram is shown in Figure 31. All we need is to find out whether (71a) is valid or not.

We assume that

$$P[I+N] = [P]_D$$

$$\text{or} \quad I+N = P^{-1}[P]_D$$

$$\text{or} \quad N = P^{-1}[P]_D - I \tag{72}$$

but the right hand side of (72), in general, does not have all diagonal elements zero which is contradictory. For example take

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}$$

$$P^{-1}[P]_D - I = \begin{bmatrix} P_{12}P_{21} & -P_{22}P_{12} \\ -P_{11}P_{21} & P_{21}P_{12} \end{bmatrix} (1/(P_{11}P_{22}-P_{12}P_{21}))$$

which does not necessary have all diagonal elements zero. So, in general, we cannot decouple a system as we want by using this configuration.

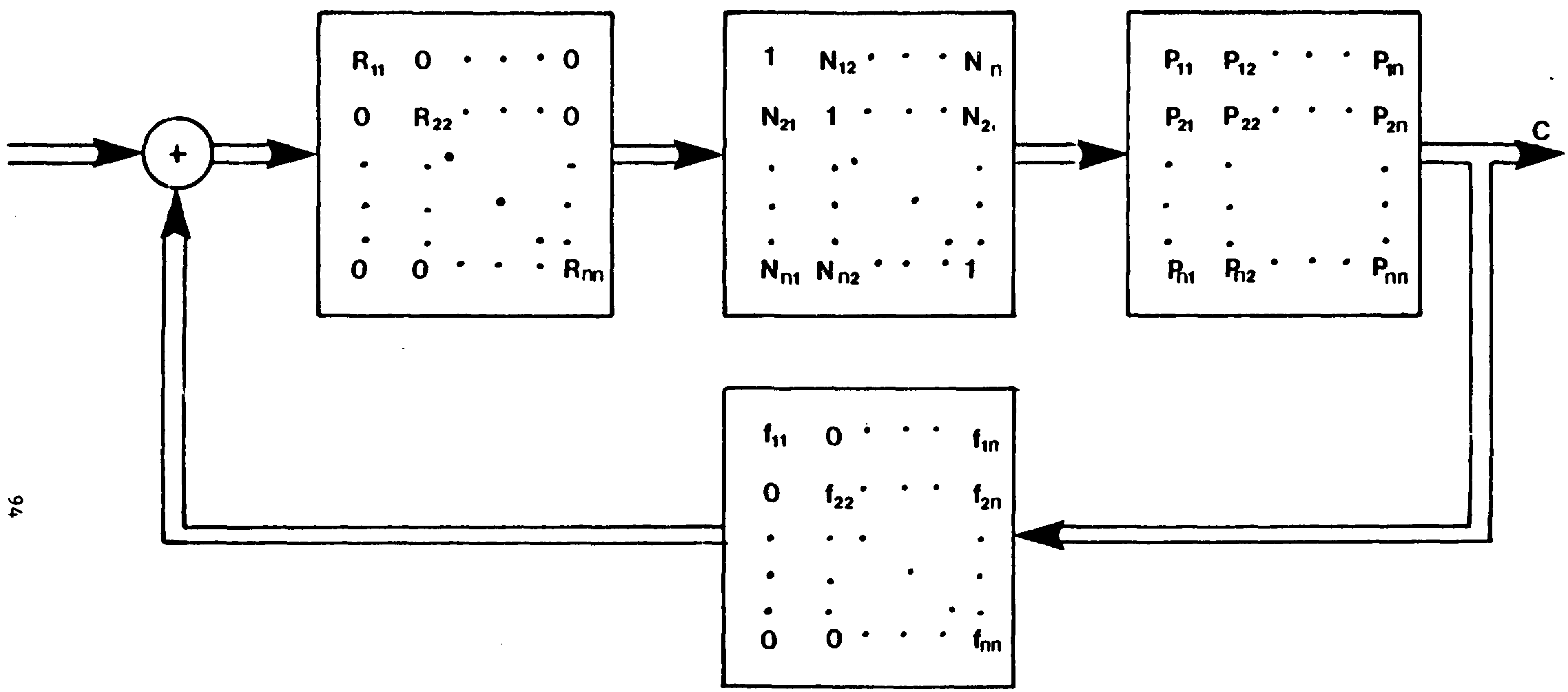

+
R11 0 0
0 R22 0
0 0 Rnn
1 N12 N n
N21 1 N2,
Nn1 Nn2 1
P11 P12 P1n
P21 P22 P2n
Pn1 Pn2 Pnn
C
f11 0 f1n
0 f22 f2n
0 0 fnn

FIGURE 31

II. P-constrained decoupler structure vs. V-constrained plant structure.

The block diagram is shown in Figure 32. From (72a), we need to prove that

$$P[\, I-N \,]^{-1} = [P]_D$$

or, $$[\, I-N \,] = [P]_D^{-1}P$$

or, $$N = I - [P]_D^{-1}P \qquad (73)$$

Since

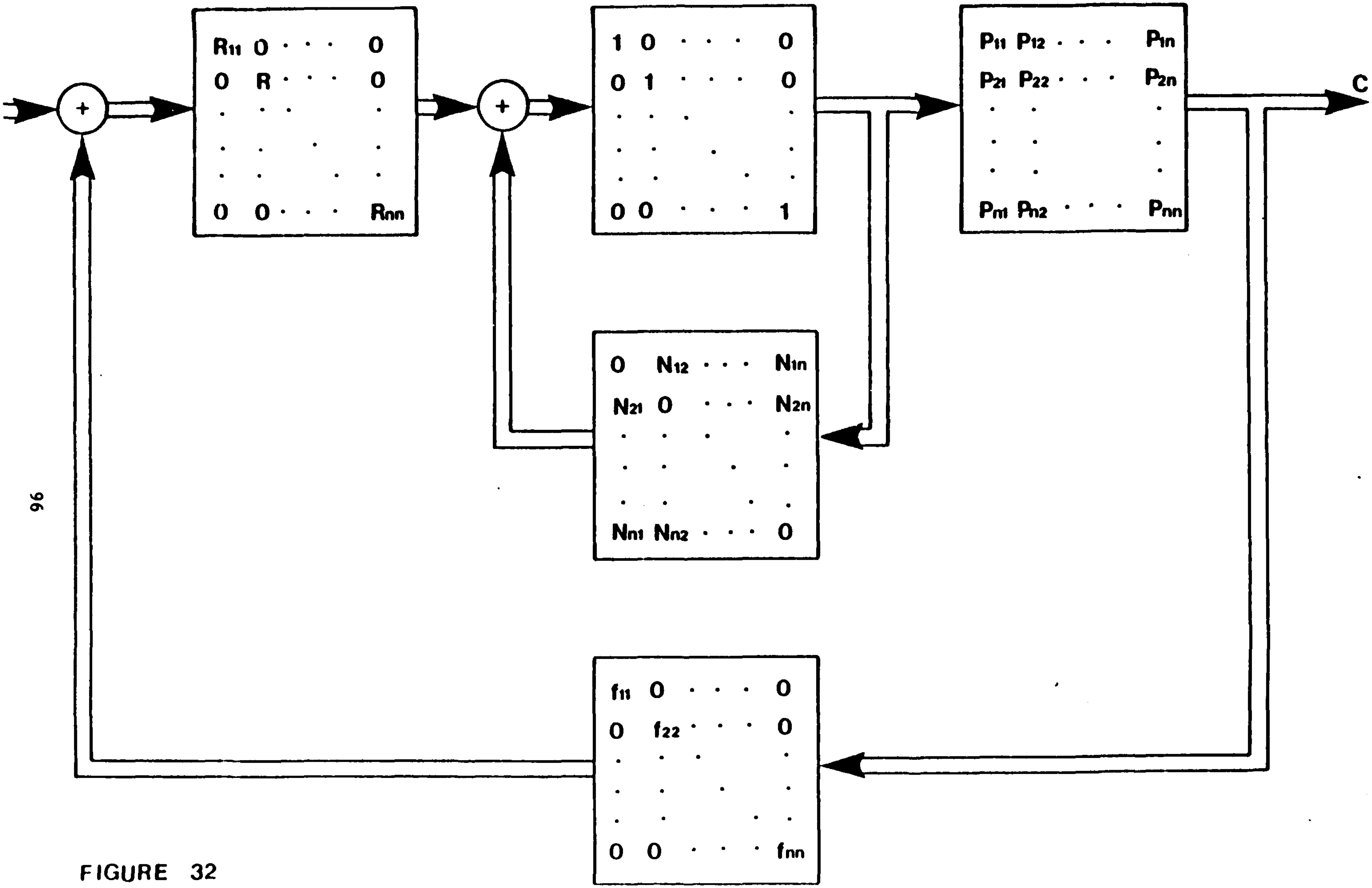
$$[P]_D^{-1} = \begin{bmatrix} 1/P_{11} & 0 & 0 \dots & 0 \\ 0 & 1/P_{22} & 0 \dots & 0 \\ . & . & & . \\ . & . & & . \\ 0 & 0 & \dots & 1/P_{nn} \end{bmatrix}$$

and $$[P]_D^{-1}P = \begin{bmatrix} 1 & P_{12}/P_{22} & \dots & P_{1n}/P_{nn} \\ P_{21}/P_{11} & 1 & \dots & P_{2n}/P_{nn} \\ . & . & & . \\ . & . & & . \\ P_{n1}/P_{11} & . & \dots & 1 \end{bmatrix}$$

Then (73) becomes

$$N = \begin{bmatrix} 0 & -P_{12}/P_{22} & \dots & -P_{1n}/P_{nn} \\ -P_{21}/P_{11} & 0 & \dots & -P_{2n}/P_{nn} \\ . & . & & . \\ . & . & & . \\ -P_{n1}/P_{11} & . & \dots & 0 \end{bmatrix}$$

which means (73) is always true and we can use this decoupling structure to accomplish our requirements.



R11 0 · · · 0
0 R · · · 0
0 0 · · · Rnn
1 0 · · · 0
0 1 · · · 0
0 0 · · · 1
P11 P12 · · · P1n
P21 P22 · · · P2n
Pn1 Pn2 · · · Pnn
C
0 N12 · · · N1n
N21 0 · · · N2n
Nn1 Nn2 · · · 0
f11 0 · · · 0
0 f22 · · · 0
0 0 · · · fnn

FIGURE 32

III. V-constrained decoupler structure vs. P-constrained plant structure.

The block diagram is given in Figure 33. From (72b), we need to prove that

$$[\ I-HK\ ]^{-1}H[\ I+N\ ] = H$$

which is equivalent to

$$P[\ I+N\ ] = H$$

or $$[\ I+N\ ] = P^{-1}H$$

and $$N = P^{-1} - I \qquad (74)$$

Since $H^{-1} = [P^{-1}]_D$

$$P^{-1}H = P^{-1}([P^{-1}]_D)^{-1} = \begin{bmatrix} 1, & P_{12}' & P_{13}', & \cdots & P_{1n}' \\ P_{21}' & 1 & P_{23}' & \cdots & P_{2n}' \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ P_{n1}' & P_{n2}' & \cdots & & 1 \end{bmatrix}$$

where P_{ij}' is some function of P_{ij}s.

which implies (74) is of the form

$$N = \begin{bmatrix} 0, & P_{12}' & P_{13}', & \cdots & P_{1n}' \\ P_{21}' & 0 & P_{23}' & \cdots & P_{2n}' \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ P_{n1}' & P_{n2}' & \cdots & & 0 \end{bmatrix}$$

It implies that (74) is always true and we can use this decoupling structure to accomplish our requirements.

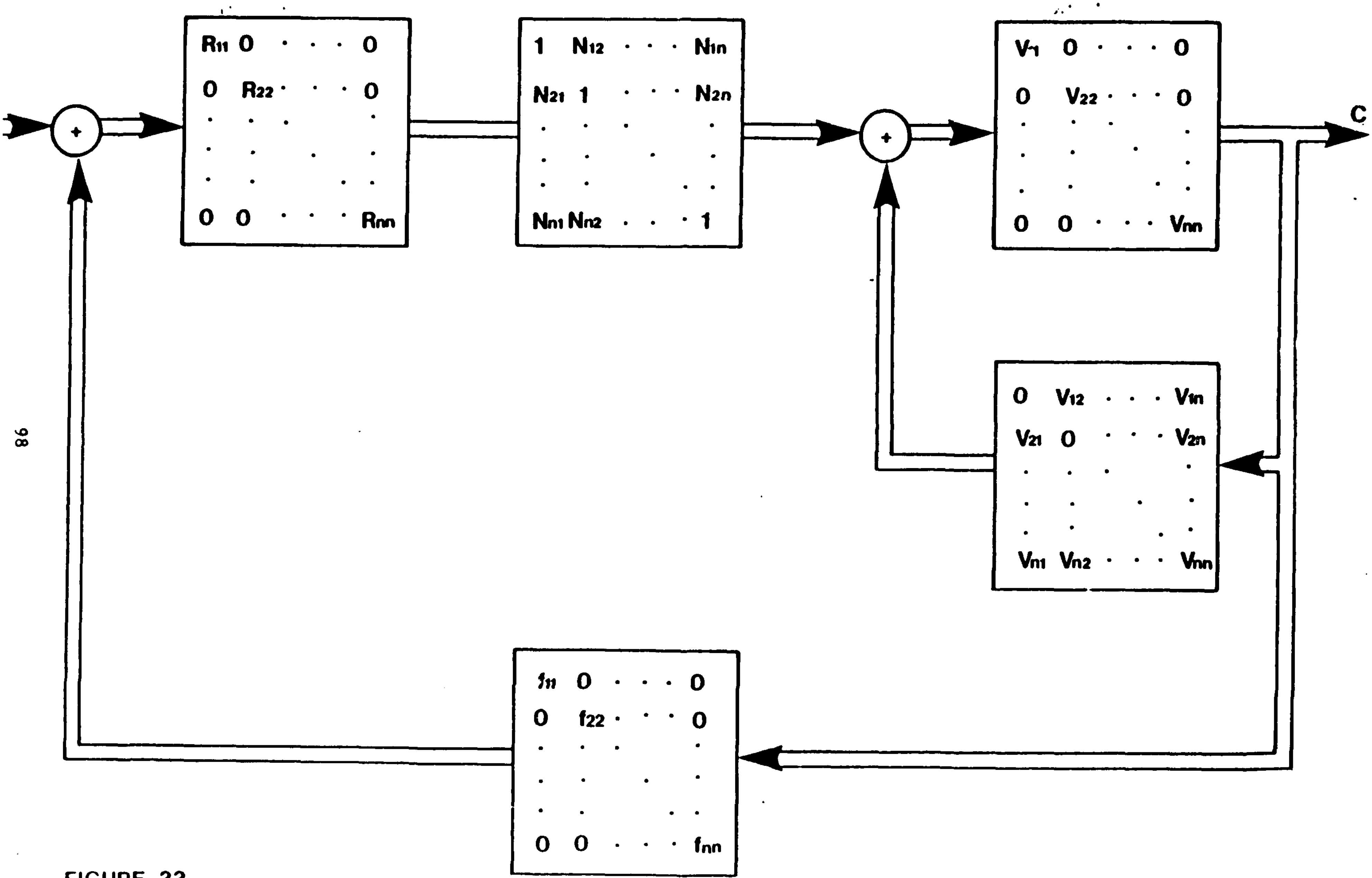


FIGURE 33

IV. V-constrained decoupler structure vs. V-constrained plant structure.

The block diagram is given in Figure 34. From (72b), we need to prove that

$$[\ I-HK\ ]^{-1}H[\ I-N]^{-1} = H$$

because $P = [\ I-HK\ ]^{-1}H$, we have

$$[\ I-N\ ]^{-1} = P^{-1}H$$

or $\quad N = I-H^{-1}P$

Since $H^{-1} = [P^{-1}]_D = [\ \mathrm{adj}\ P\ ]_D/(\det P)$

and let the cofactors be A_{ij}, then

$$H^{-1}P = (1/\det P)\begin{bmatrix} P_{11}A_{11} & P_{12}A_{12} & \cdots & P_{1n}A_{1n} \\ P_{12}A_{12} & P_{22}A_{22} & \cdots & P_{2n}A_{2n} \\ . & . & & . \\ . & . & & . \\ P_{n1}A_{n1} & P_{n2}A_{n2} & \cdots & P_{nn}A_{nn} \end{bmatrix}$$

As we known

$$\det P = \sum_{i=1}^{n} P_{ij}A_{ij} = \sum_{j=1}^{n} P_{ij}A_{ij}$$

In general, we have $(P_{ij}A_{ij})/(\det P) \neq 1$, for $i=1,\ldots,n$.

That means, $[\ I-H^{-1}P]_D$ are not all zeros. It implies that, in general, we cannot decouple a system as we want by using this configuration.

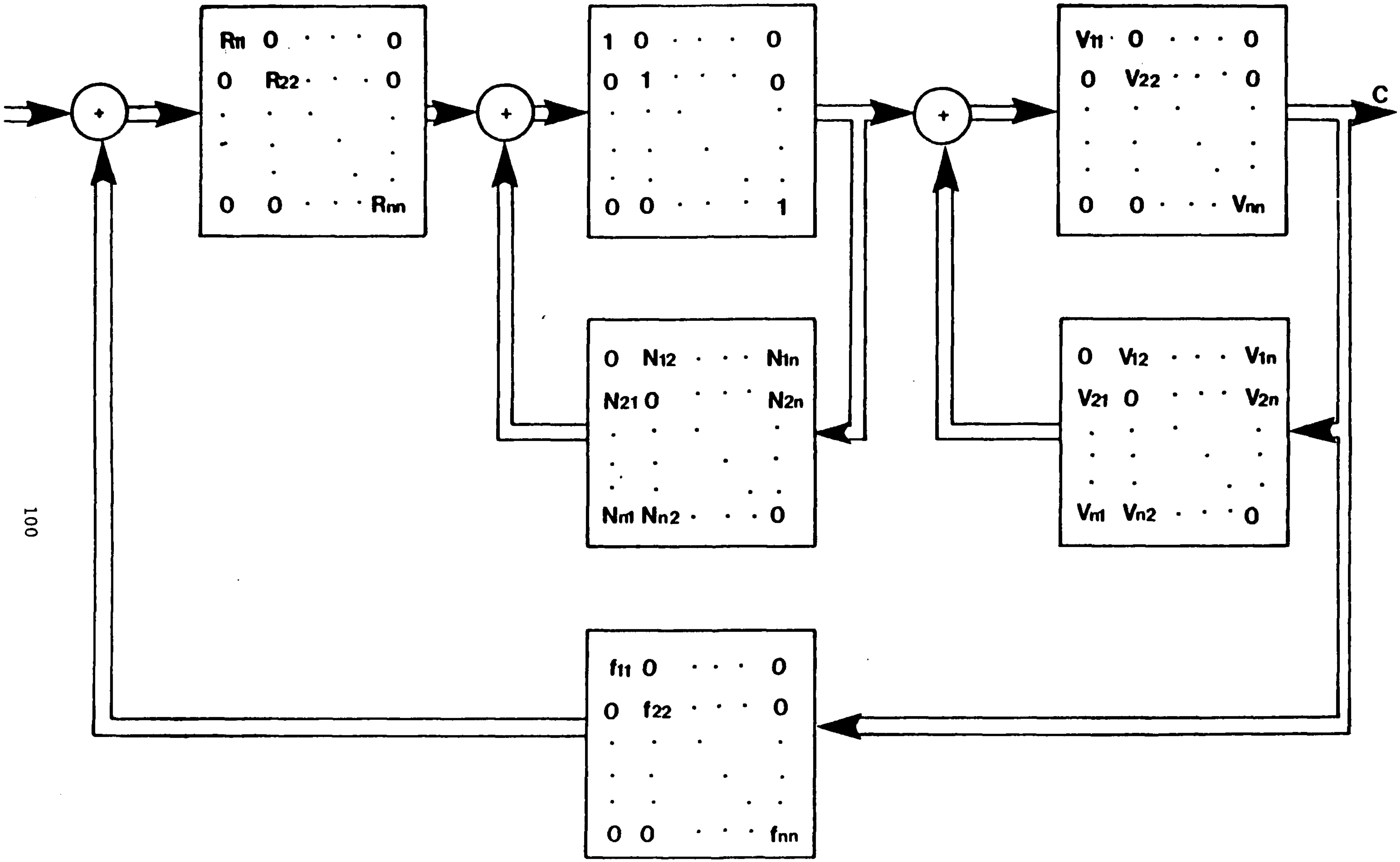


FIGURE 34

We can conclude here that , from our proofs, when a P-constrained plant model is used, we need a V-constrained decoupler to fulfill all our requirements, or vice verse.

As we mentioned, from only the inputs and the outputs of a MIMO system, there is no way to identify whether it is P- or V- constrained. We need to assume that the structure of the MIMO system has been determined by some physical means, and once this mathematical model is at hand, the results of this section can be applied. Notice that, for example, when we have a P-constrained plant structure, the most effective way to decouple is to use a V-constrained decoupler and the parameters are specified by (74). But, for certain reasons, we must use a P-constrained decoupler structure, we can transform the mathematical model of the plant into a V-constrained one and use this identified structure to specify the P-constrained decoupler parameters by (73).

5.0 Multivariable Dead-Time Compensator

In this section we will generalize the well known Smith compensator to the multivariable case.

5.1 Smith Compensator

Consider the SISO time-delayed system shown in Figure 35, where $R_1(s)$ and $C_1(s)$ are the Laplace transform of the input and output functions. $T_R(s)$, $T_P(s)$, and $T_F(s)$ are the transfer functions of the input compensator, the plant, and the feedback compensator respectively. τ is the time delay of the plant. The transfer function of the block diagram shown in Figure 35 is

$$T_1(s) = \frac{C_1(s)}{R_1(s)} = \frac{T_R(s)T_P(s)e^{-\tau s}}{1 + T_F(s)T_R(s)T_P(s)e^{-\tau s}} \tag{75}$$

Let $T_R(s) = 1/(s+3)$; $T_P(s)e^{-\tau s} = e^{-0.2s}/[(s+1)(s+2)]$; $T_F(s) = 1/s$,

then

$$T_1(s) = \frac{\dfrac{1}{(s+3)}\,\dfrac{1}{(s+1)(s+2)}\,e^{-0.2s}}{1 + \dfrac{1}{(s+3)}\,\dfrac{1}{(s+1)(s+2)}\,e^{-0.2s}}$$

$$= \frac{s\,e^{-0.2s}}{(s)(s+1)(s+2)(s+3) + e^{-0.2s}} \tag{76}$$

The existence of the time-delay in the denominator of $T_1(s)$, as seen from the example, is a problem of systems stability. Now, consider the block diagram

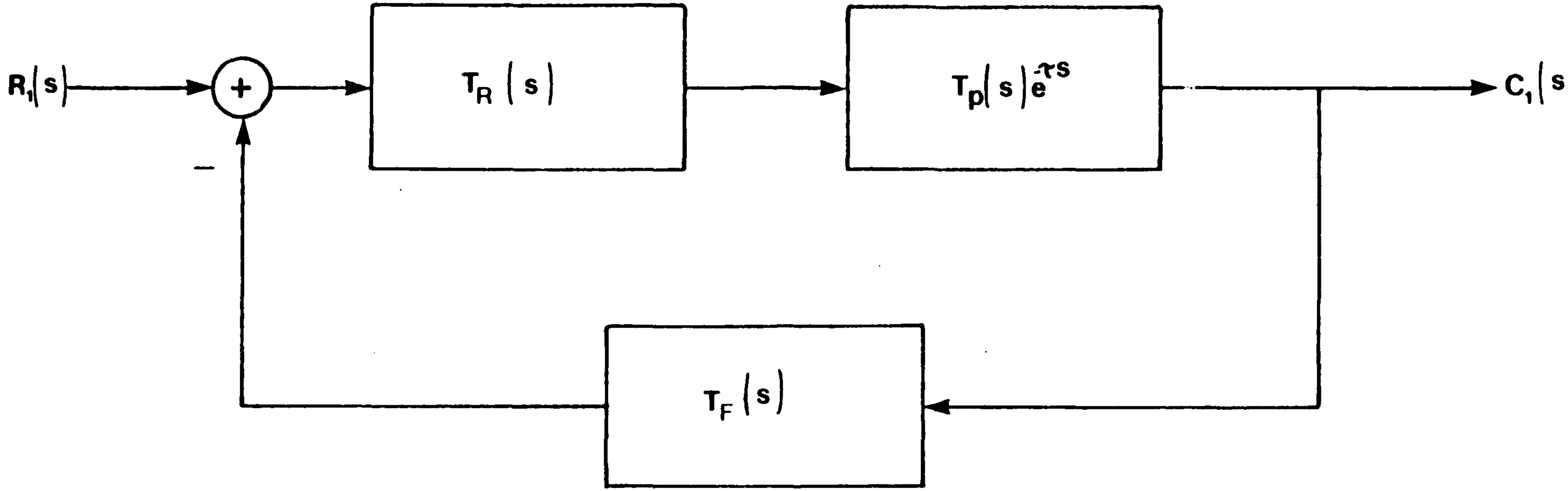


FIGURE 35

shown in Figure 35 with a smith compensator added, which is redrawn in Figure 36.

The transfer function of the block diagram shown in Figure 36 is

$$T_2(s) = \frac{C_2(s)}{R_2(s)} = \frac{\dfrac{T_R(s)}{1 + T_P(s)T_R(s)T_F(s)(1-e^{-\tau s})} T_P(s)e^{-\tau s}}{1 + \dfrac{T_F(s)T_R(s)}{1 + T_P(s)T_R(s)T_F(s)(1-e^{-\tau s})} T_P(s)e^{-\tau s}}$$

$$= \frac{T_R(s)T_P(s)e^{-\tau s}}{1 + T_F(s)T_R(s)T_P(s)e^{-\tau s}} \qquad (77)$$

With the example's numerical values,

$$T_2(s) = \frac{\dfrac{1}{(s+3)} \dfrac{1}{(s+1)(s+2)} e^{-0.2s}}{1 + \dfrac{1}{s} \dfrac{1}{(s+3)} \dfrac{1}{(s+1)(s+2)} e^{-0.2s}}$$

$$= \frac{s\ e^{-0.2s}}{(s)(s+1)(s+2)(s+3) + 1} \qquad (78)$$

Examine the denominator of $T_2(s)$, we find that there is no time-delay in it. That is, the problem of system stability raised from the time-delay is solved.

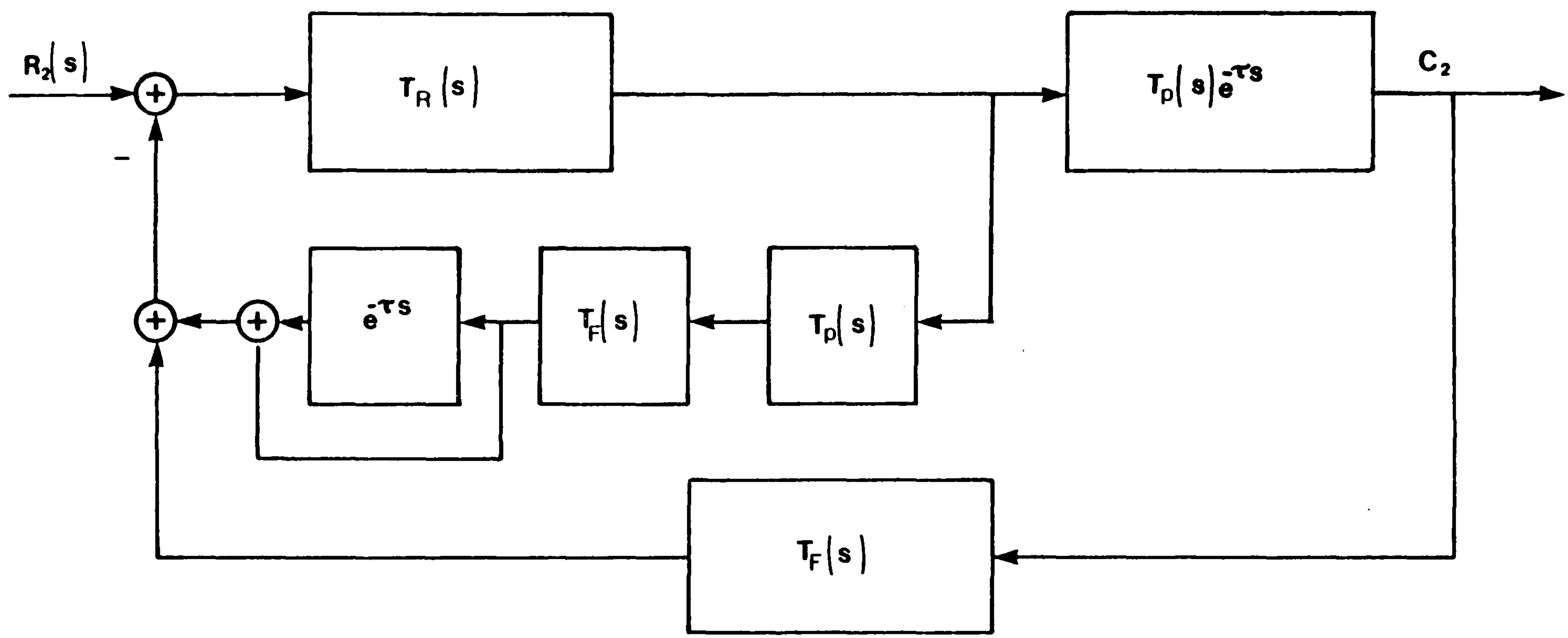


FIGURE 36

5.2 Multivariable Smith Compensator

To generalize the Smith compensator structure to the multivariable cases, we need to examine Figure 36 in more details. If we simplify the Smith compensator structure shown in the Figure, we can have the simplified block diagram shown in Figure 37.

The input and output of the Smith compensator is marked with X(s) and W(s) respectively. The transfer function

$$S_H(s) = \frac{W(s)}{X(s)} = (1-e^{-\tau s})\ T_F(s)T_P(s)$$

$$= T_F(s)T_P(s) - T_F(s)T_P(s)e^{-\tau s} \qquad (79)$$

Notice that the first term of (79) is the product of the transfer functions of the plant and the feedback compensator without the time-delay while the second term is the same product with the time-delay.

For the multivariable case, let the transfer matrices of the plant and the feedback compensator are P and F respectively. To generalize further, we assume that there are different time-delays in the matrices P and F. If the transfer matrices with no time-delays (means all the τ's are zero) are P^* and F^*, then the multivariable Smith compensator will have the structure

$$S_H = F^*P^* - FP \qquad (80)$$

notice that (80) is a transfer matrices equation. To elaborate, consider the multivariable system shown in Figure 38. Where R, P, and F are the transfer

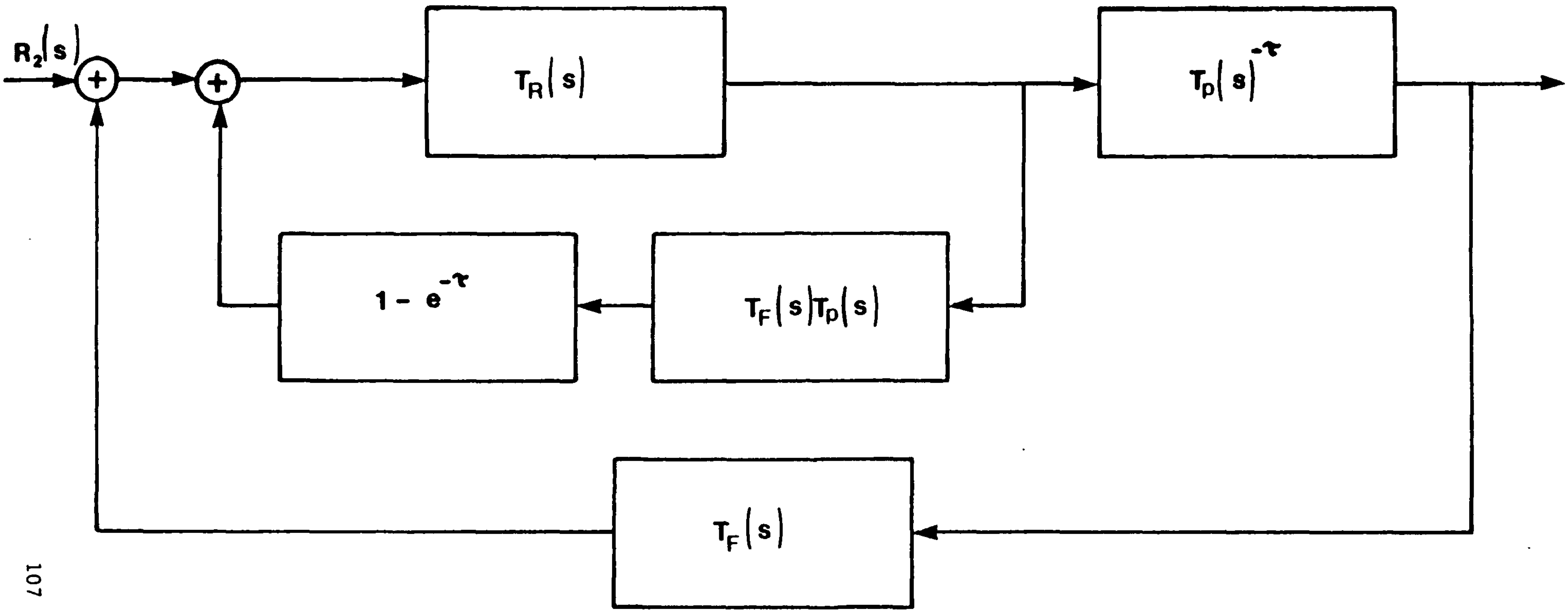


FIGURE 37

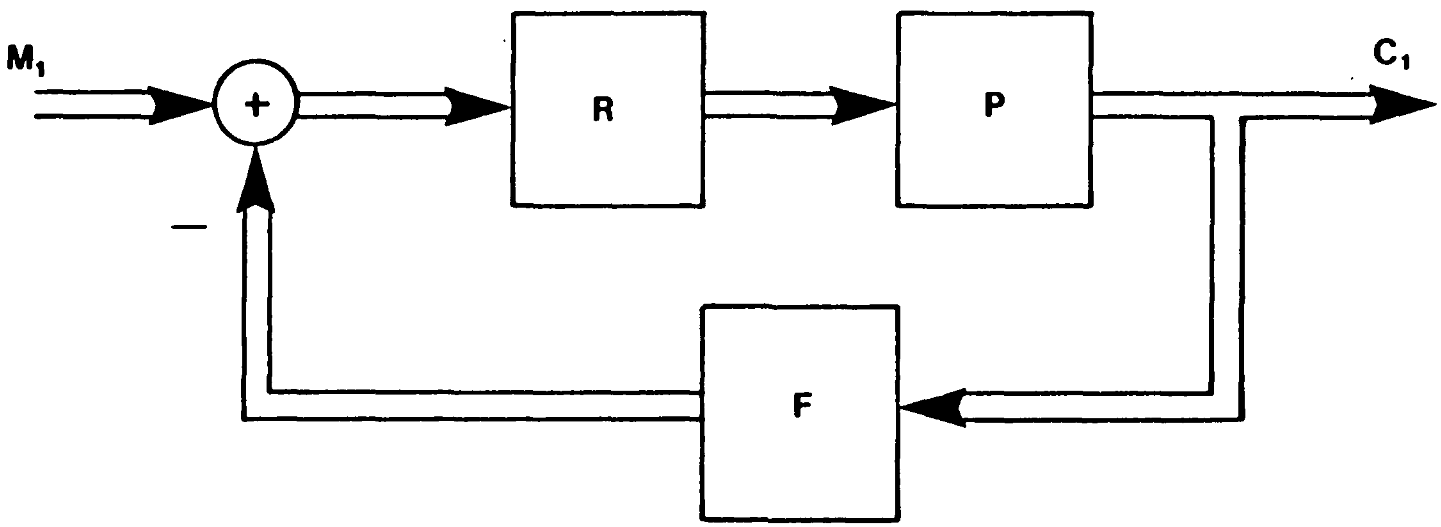


FIGURE 38

matrices of the input compensator, the plant, and the feedback compensator respectively. $M_1^T = [\ M_{11}(s),\ M_{12}(s),\ \ldots,\ M_{1n}(s)\ ]$ is the Laplace transform of the input functions while $C_1^T = [\ C_{11}(s),\ C_{12}(s),\ \ldots,\ C_{1n}(s)\ ]$ is that of the output functions. Consider an numerical example with

$$P = \begin{bmatrix} \dfrac{1}{(s+1)} e^{-0.1s} & 0 \\ \dfrac{1}{(s+3)} & \dfrac{1}{(s+2)} e^{-0.5s} \end{bmatrix}$$

$$R = \begin{bmatrix} \dfrac{1}{(s)} & 0 \\ 0 & \dfrac{1}{(s+3)} \end{bmatrix}$$

$$F = \begin{bmatrix} e^{-0.2s} & 0 \\ 0 & e^{-0.3s} \end{bmatrix}$$

Since $C_1 = [\ I+PRF\ ]^{-1} PRM_1$ (81)

$$PR = \begin{bmatrix} \dfrac{e^{-0.1s}}{(s)(s+1)} & 0 \\ \dfrac{1}{(s)(s+3)} & \dfrac{e^{-0.5s}}{(s+2)(s+3)} \end{bmatrix}$$

$$PRF = \begin{bmatrix} \dfrac{e^{-0.3s}}{(s)(s+1)} & 0 \\ \dfrac{e^{-0.2s}}{(s)(s+3)} & \dfrac{e^{-0.8s}}{(s+2)(s+3)} \end{bmatrix}$$

Using Grassmann algebra with

$$c_1^T = [\ 1,0\ ],\ c_2^T = [\ 0,1\ ],$$

$$b_1 = \begin{bmatrix} \dfrac{e^{-0.1s}}{(s)(s+1)} \\ \\ \dfrac{1}{(s)(s+3)} \end{bmatrix},\quad b_2 = \begin{bmatrix} 0 \\ \\ \dfrac{e^{-0.5s}}{(s+2)(s+3)} \end{bmatrix}.$$

We have

$$C_1 = \begin{bmatrix} \dfrac{e^{-0.1s}}{(s)(s+1)+e^{-0.3s}} & 0 \\ \\ \dfrac{(s+2)[s(s+1)+e^{-0.3s}-e^{-0.1s}]}{(s)[s(s+1)+e^{-0.3s}][(s+2)(s+3)+e^{-0.8s}]} & \dfrac{e^{-0.5s}}{(s+2)(s+3)+e^{-0.8s}} \end{bmatrix} \tag{82}$$

Examine (82), we find that there are time-delays in the denominators. In general, the characteristic equation of (81),

$$\det [\ I+PRF\ ] = 0$$

contains time-delays if matrices P and F have time-delays. When multivariable Smith compensator (80) is used, the block diagram is shown in Figure 39. To illustrate, use the numerical example again, then (80) becomes

$$S_F = \begin{bmatrix} 1 & 0 \\ \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \dfrac{1}{(s+1)} & 0 \\ \\ \dfrac{1}{(s+3)} & \dfrac{1}{(s+2)} \end{bmatrix} - \begin{bmatrix} e^{-0.2s} & 0 \\ \\ 0 & e^{-0.3s} \end{bmatrix} \begin{bmatrix} \dfrac{e^{-0.1s}}{(s+1)} & 0 \\ \\ \dfrac{1}{(s+3)} & \dfrac{e^{-0.5s}}{(s+2)} \end{bmatrix}$$

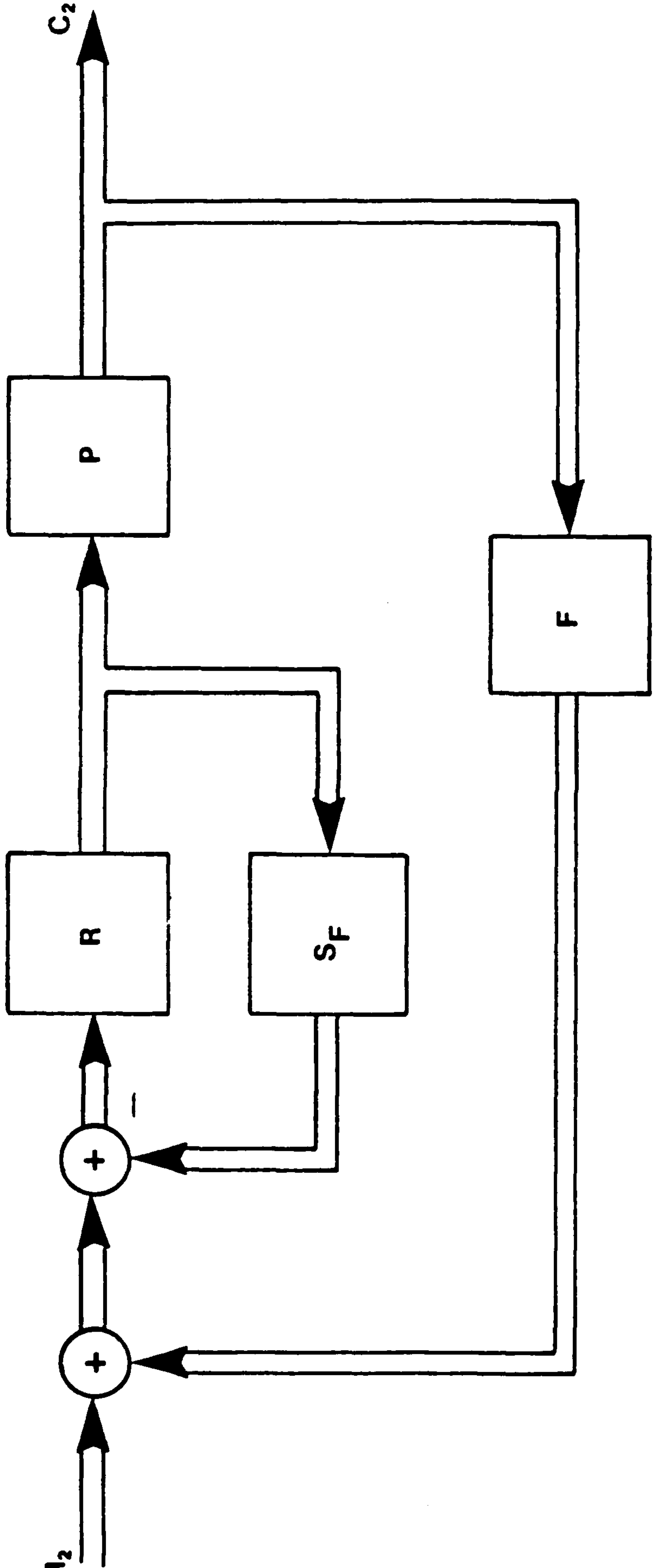


FIGURE 39

$$= \begin{bmatrix} \frac{1}{(s+1)}(1 - e^{-0.3s}) & 0 \\ \frac{1}{(s+3)}(1 - e^{-0.3s}) & \frac{1}{(s+2)}(1 - e^{-0.8s}) \end{bmatrix}$$

Simplify the inner loop of Figure 39, we have the resultant input compensator

$$R^* = [\ I+RS_F\ ]^{-1}R$$

Using Grassmann algebra with

$$c_1^T = [\ 1,0\ ],\ c_2^T = [\ 0,1\ ],$$

$$b_1 = \begin{bmatrix} \frac{1}{(s)} \\ 0 \end{bmatrix} , \quad b_2 = \begin{bmatrix} 0 \\ \frac{1}{(s+3)} \end{bmatrix}$$

let $d_1 = (s)(s+1) + (1-e^{-0.3s})$,

and $d_2 = (s+2)(s+3) + (1-e^{-0.8s})$.

Then

$$R^* = \begin{bmatrix} \frac{(s+1)}{d_1} & 0 \\ \frac{-(s+1)(s+2)(1-e^{-0.3s})}{(s+3)\ d_1 d_2} & \frac{(s+2)}{d_2} \end{bmatrix}$$

Proceed as the previous developments with R^* as the input compensator instead of R, we have

$$C_2 = [\, I + PR^*F \,]^{-1} PR^*M_2$$

let $d_3 = (s+2)(s+3) + (1-e^{-0.5s})$

then

$$PR^* = \begin{bmatrix} \dfrac{e^{-0.1s}}{d_1} & 0 \\ \dfrac{(s+1)}{(s+3)} \dfrac{d_3}{d_1 d_2} & \dfrac{e^{-0.5s}}{d_2} \end{bmatrix}$$

$$PR^*F = \begin{bmatrix} \dfrac{e^{-0.3s}}{d_1} & 0 \\ \dfrac{(s+1)}{(s+3)} \dfrac{d_3}{d_1 d_2} & \dfrac{e^{-0.8s}}{d_2} \end{bmatrix}$$

Using Grassmann algebra with

$c_1^T = [\, 1,0 \,]$, $c_2^T = [\, 0,1 \,]$,

$$b_1 = \begin{bmatrix} \dfrac{e^{-0.1s}}{d_1} \\ \dfrac{(s+1)}{(s+3)} \dfrac{d_3}{d_1 d_2} \end{bmatrix}, \quad b_2 = \begin{bmatrix} 0 \\ \dfrac{e^{-0.5s}}{d_2} \end{bmatrix}.$$

we have

$$C_2 = \begin{bmatrix} \dfrac{e^{-0.1s}}{(s)(s+1)+1} & 0 \\ \dfrac{(s+1)[(s+2)(s+3)+1-e^{-0.5s}]}{(s+3)[\ s(s+1)+1\ ][\ (s+2)(s+3)+1\ ]} & \dfrac{e^{-0.5s}}{(s+2)(s+3)+1} \end{bmatrix} \tag{83}$$

Comparing with (82), we find that there are no time-delays in all the denominators of (83). This is because we use a multivariable Smith compensator as shown in Figure 39 and equation (80). A general prove of (80) is developed here now.

Refer to Figure 39 again, but now $M_2^T=[M_{21}(s),\dots,M_{2n}(s)]$ and $C_2^T=[C_{21}(s),\dots,C_{2n}(s)]$. R, P, and F are $n \times n$ transfer matrices. We assume that there are time-delays in matrices P and F only. Let P* and F* are matrices of P and F without the delays respectively. We want to prove that by using (80), the characteristic equation of the resultant transfer matrix from Figure 39 consists of no time-delays.

Writing $R^* = [\ I+RS_F\ ]^{-1}R = T^{-1}R$, then Figure 39 is equivalent to Figure 40. And

$$\begin{aligned} C_2 &= [\ I+PR^*F\ ]^{-1}PR^*M_2 \\ &= [\ I+PT^{-1}RF\ ]^{-1}PT^{-1}RM_2 \end{aligned}$$

Since $\quad [\ I+PT^{-1}RF\ ] = P[\ I+T^{-1}RFP\ ]P^{-1}$

$$= PT^{-1}[\ T+RFP\ ]P^{-1}$$

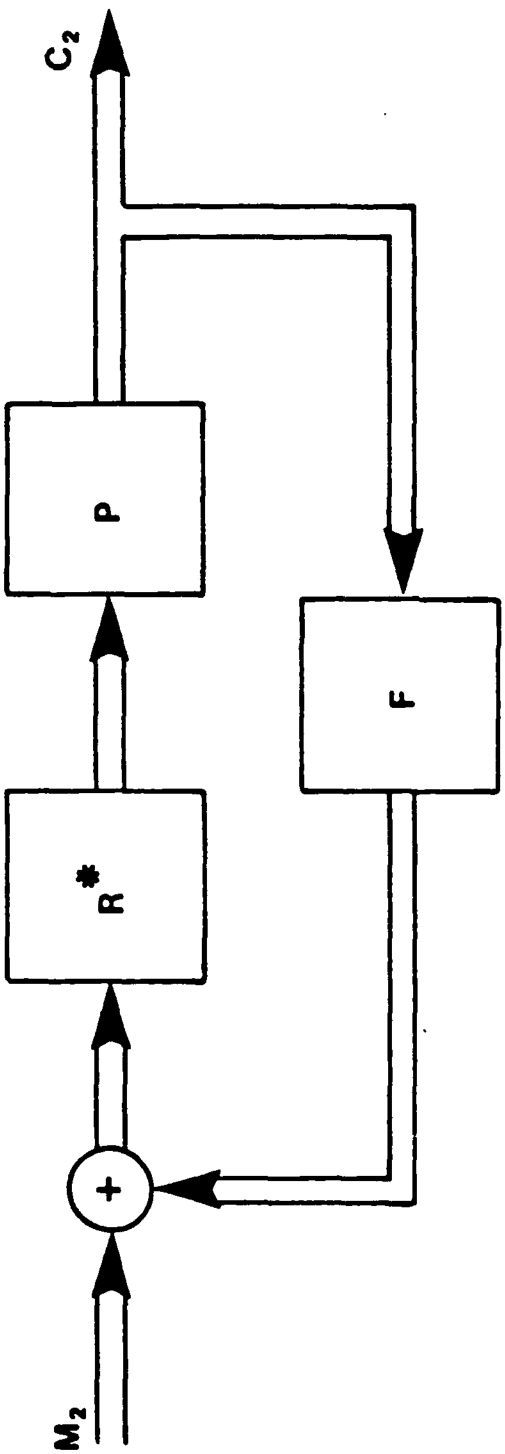


FIGURE 40

implies that

$$[\ I+PT^{-1}RF\ ]^{-1} = P[\ T+RFP\ ]^{-1}TP^{-1}$$

Therefore

$$\begin{aligned} C_2 &= [\ I+PT^{-1}RF\ ]^{-1}PT^{-1}RM_2 \\ &= P[\ T+RFP\ ]^{-1}TP^{-1}PT^{-1}RM_2 \\ &= P[\ T+RFP\ ]^{-1}RM_2 \end{aligned}$$

As $\quad T = I + RS_F$

$$= I + R[\ F^*P^* - FP\ ]$$

we have $C_2 = P[\ I+RF^*P^*\ ]^{-1}RM_2$

The characteristic equation of this multivariable system is

$$\det [\ I+RF^*P^*\ ] = 0$$

by our assumption, R, F^*, and P^* consist of no time-delays, which implies that the characteristic equation consists no time-delays. And, we have proved that (80) is a multivariable Smith compensator.

6.0 More Applications of Grassmann Algebra.

In this chapter, we will present two more applications of the Grassmann algebra. The algebra is now used as an intermediate step to obtain simultaneous equations instead of the symbolic evaluations of determinants as the previous chapters

6.1 Liapunov Equation.

Either in stability studies, or in the ouput feedback optimal control, a fundamental symbolic matrix must be solved. That is the Liapunov equation:

$$AP + AP^T = -Q \qquad (84)$$

where A is the system matrix of general *linear* equation

$$\dot{x} = Ax + bu \qquad (85)$$

and P is a symmetrical matrix whose parameters are usually to be determined.

Grassmann algebra is most helpful in the investig.tion. A general derivation is shown as follows.

The frequently used technique is to expand (84) into a set of simultaneous equations by using the Knonecker product

$$[\ A^T \otimes I + I \otimes A^T\ ]\ [p] = -[q] \qquad (86)$$

where $\otimes$ means Knonecker product, the matrices in detail are as follows:

$$[A^T \otimes I + I \otimes A^T] = \begin{bmatrix} a_{11}I & a_{21}I & \dots & a_{n1}I \\ a_{12}I & a_{22}I & \dots & a_{n2}I \\ . & . & & . \\ . & . & & . \\ a_{1n}I & a_{2n}I & \dots & a_{nn}I \end{bmatrix} + \begin{bmatrix} A^T & 0 & \dots & 0 \\ 0 & A^T & \dots & 0 \\ . & . & & . \\ . & . & & . \\ 0 & 0 & \dots & A^T \end{bmatrix} \qquad (87)$$

If A is transformed into the phase variable form by using the Krylov transformation, e.g (87) will be simplified to

$$[A^T \otimes I + I \otimes A^T] = \begin{bmatrix} 0 & 0 & \dots & 0 & a_n I \\ I & 0 & \dots & 0 & a_{n-1}I \\ . & . & & . & . \\ . & . & & . & . \\ 0 & 0 & \dots & I & a_1 I \end{bmatrix} + \begin{bmatrix} A^T & 0 & \dots & 0 \\ 0 & A^T & \dots & 0 \\ . & . & & . \\ . & . & & . \\ 0 & 0 & \dots & A^T \end{bmatrix} \qquad (88)$$

and $p = [\, p_{11}, p_{12}, \dots, p_{1n}, p_{12}, p_{22}, \dots, p_{2n}, p_{13}, \dots, p_{1n}, p_{2n}, \dots, p_{nn} \,]$ or arrange the columns of P continuously into a vector; q is defined similarly.

Then the Liapunov matrix equation (84) becomes the following:

$$W\,p = -q \qquad (89)$$

Equation (89) is $n \times n$ algebraic equations

If $P_{ij} = P_{ji}$, $(i \neq j)$, or P is a symmetrical matrix. There are only

$$\frac{n(n-1)}{2} \tag{90}$$

independent equations and the same number unknowns

$$W\, p_r = -q_r$$

where the dimension r is equal to the shown in (90)

In detail, we have

$$W \begin{bmatrix} P_{11} \\ P_{12} \\ P_{13} \\ \cdot \\ \cdot \\ P_{nn} \end{bmatrix} = -\begin{bmatrix} Q_{11} \\ Q_{12} \\ Q_{13} \\ \cdot \\ \cdot \\ Q_{nn} \end{bmatrix} \tag{91}$$

How to find W by the computer is elegently established as follows:

Assume $W = [\, W_{rs} \,],$ $\qquad A = [\, a_{ij} \,]$ (92)

Let $I = 1,2,\ldots,n; \qquad J = I,(I+1),\ldots,n$

$K = 1,2,\ldots,n; \qquad L = K,(K+1),\ldots,n$ (93)

$$\begin{array}{c|c} & (K,L) \\ \hline (I,J) & W_{rs} \end{array}$$

$$K = I \begin{cases} L \neq J & \begin{cases} I=J \longrightarrow W(r,s) = 2A(L,J) \\ I \neq J \longrightarrow W(r,s) = A(L,J) \end{cases} \\ & \\ L = J & \begin{cases} K=J \longrightarrow W(r,s) = 2A(I,I) \\ K \neq J \longrightarrow W(r,s) = A(I,I)+A(J,J) \end{cases} \end{cases}$$

$$K = I \begin{cases} L = J & \begin{cases} I=J \longrightarrow W(r,s) = 2A(L,J) \\ I \neq J \longrightarrow W(r,s) = A(L,J) \end{cases} \\ & \\ L \neq J & \begin{cases} K=J \longrightarrow W(r,s) = 2A(I,I) \\ K \neq J \begin{cases} L=I \longrightarrow W(r,s) = A(k,J) \\ L \neq I \longrightarrow W(r,s) = 0 \end{cases} \end{cases} \end{cases}$$

The algorithm shown above is much simpler than previous works.

When the Grassmann algebra is applied to find W symbolically and A is restricted to the companion form. We have the following results for n=2

$$A = \begin{bmatrix} 0 & 1 \\ -a_2 & -a_1 \end{bmatrix}$$

$$W = A^T \otimes I + I \otimes A^T = \begin{bmatrix} 0 & -a_2 & -a_2 & 0 \\ 1 & -a_1 & 0 & -a_2 \\ 1 & 0 & -a_1 & -a_2 \\ 0 & 1 & 1 & -2a_1 \end{bmatrix}$$

W in Grassmann terminology is

-2134
+2143
-2413
-3142
-4214
+3412

For n=3 where

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_3 & -a_2 & -a_1 \end{bmatrix}$$

and W in Grassmann terminology is as follows

+ 231564789
- 231564798
+ 231564897
+ 231569487
+ 231584796
+ 231586497
+ 231764859
+ 231765489
- 231765498
+ 231769458
+ 231785496
+ 231786459
- 231789456
+ 236584197
+ 236784159
+ 251764893
+ 251769483
+ 251786493
- 253764189
+ 253764198
- 253784196
+ 256784193
- 431569728
+ 431569827
- 431586729
+ 431589726
+ 431765829
+ 432569187
+ 432586197
+ 432765189
- 432765198
+ 432769158
+ 432785196
+ 432786159
- 432789156
+ 436589127
+ 436785129
+ 451769823
+ 452769183
+ 452786193
+ 453769128
+ 453786129
- 453789126
+ 456789123

6.2 Riccati Equation.

The matrix Riccati equation is the core for solving the optimal problem of a linear plant with respect to the quadratic performance index. There are two schools of attack: one is to consider the matrix Liapunov equation and based on the successive Liapunov equation solutions to approach the matrix Riccati equation. The well known Kleinman technique belongs to this school. However, Kleinman's method is purely numerical and iterative: every trial should start from the very beginning. The other school is to expand the matrix Riccati equation into a set of simultaneous equations and keep the symbolic parameters as far as possible. This section offers an expansion technique in the second school. The method is to establish algorithms in order to use Grassmann algebra after the set of simultaneous equations formed.

consider a linear plant,

$$\dot{x} = Ax + Bu \tag{94}$$

The performance index is:

$$J = \frac{1}{2} \int_0^\infty (x^T QX + u^T Ru)dt \tag{95}$$

Where Q and R are symmetric matrices.

The fundamental problem is to solve the following Riccati equation:

$$\dot{P} = PBR^{-1}B^T P - (PA+A^T P) - Q \tag{96}$$

with using the relation

$$\phi = Px \tag{97}$$

we obtain the control law

$$u = -R^{-1}B^{T}\phi(t) \tag{98}$$

It is natural to think that we expand (96) into a set of simultaneous equations:

$$p = \Omega\, p - \alpha\, p - q \tag{99}$$

where

$$p = [\ p_{11}, p_{12}, p_{13}, \cdots, p_{1n}, p_{22}, p_{23}, \cdots, p_{n-1,n}, p_{n,n}\ ]^{T} \tag{100}$$

and

$$q = [\ q_{11}, q_{12}, q_{13}, \cdots, q_{1n}, q_{22}, q_{23}, \cdots, q_{n-1,n}, q_{n,n}\ ]^{T} \tag{101}$$

we would like to construct two algorithms in order to find the corresponding matrices Ω and α. where

$$\alpha = [\ a_{rs}\ ] \qquad r,s=1,2,\ldots,n(n+1)/2 \tag{102}$$

$$\Omega = [\ a_{rs}\ ] \qquad r=1,2,\ldots,n(n+1)/2$$

$$s=1,2,\ldots,n(n+1)(n^{2}+n+2)/8 \tag{103}$$

(I) Formula for finding matrix α. (APL function: AFA)

$$\alpha = [\ \alpha_{rs}\ ], \qquad r,s=1,2,\ldots,n(n+1)/2$$

The values of α_{rs} are found by two steps.

(1) form a matrix V by writting the subscripts of the elements of the vector p. (APL function: INDEX)

$$V = [\ v_{ij}\ ] = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ . & . \\ . & . \\ . & . \\ 1 & n \\ 2 & 2 \\ 2 & 3 \\ . & . \\ . & . \\ . & . \\ n-1 & n-1 \\ n-1 & n \\ n & n \end{bmatrix} \qquad p = \begin{bmatrix} P_{11} \\ P_{12} \\ . \\ . \\ . \\ P_{1n} \\ P_{22} \\ P_{23} \\ . \\ . \\ . \\ P_{n-1,n-1} \\ P_{n-1,n} \\ P_{n,n} \end{bmatrix}$$

Let $I = v_{r1}$, $J = v_{r2}$

and $K = v_{s1}$, $L = v_{s2}$

(2) Define a function:

$$f(x,y) = \begin{cases} 1 & (x=y) \\ 0 & (x \neq y) \end{cases}$$

Let $a = f(I,K)$, $b = f(I,L)$

and $c = f(J,K)$, $d = f(J,L)$

and $T = 1 + f(I,J)$

Then the elements α_{rs} is evaluated by the following formula:

$$\alpha_{rs} = T\{dA(K,I)+a(1-bd)A(L,J)+(1-a)(1-d)[cA(L,I)+b(1-c)A(K,J)]\} \tag{104}$$

To illustrate the steps shown above, we give a simple example as follows:

If A is a 2×2 matrix

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

then α is a 3×3 matrix because $n(n+1)/2 = 3$,

$$\alpha = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} \\ \alpha_{21} & \alpha_{22} & \alpha_{23} \\ \alpha_{31} & \alpha_{32} & \alpha_{33} \end{bmatrix}$$

We write the matrix V first:

$$V = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 2 & 2 \end{bmatrix}$$

$r=1$: $I=v_{11}=1$, $J=v_{12}=1$, $T=1+f(I,J)=1+f(1,1)=1+1=2$

$s=1$: $K=v_{11}=1$, $L=v_{11}=1$

$a=b=c=d=f(1,1)=1$

$\therefore\ 1-bd=1-a=1-d=1-c=0$

substituting the values of a, b, c, d and T into (104), we have

$\alpha_{11}=TdA(K,I)=2A(1,1)=2A_{11}$

$s=2$: $K=v_{21}=1$, $L=v_{22}=2$

$a=f(I,K)=f(1,1)=1$, $b=f(I,L)=f(1,2)=0$

$c=f(J,K)=f(1,1)=1$, $d=f(J,L)=f(1,2)=0$

$\therefore\ 1-bd=1$, $1-a=0$

$\therefore\ \alpha_{12}=Ta(1-bd)A(L,J)=2A(2,1)=2A_{21}$

$s=3$: $K=v_{31}=2$, $L=v_{32}=2$

$a=f(I,K)=f(1,2)=0$, $b=f(I,L)=f(1,2)=0$

$c=f(J,K)=f(1,2)=0$, $d=f(J,L)=f(1,2)=0$

$\therefore\ \alpha_{13}=0$

$r=2$: $I=v_{21}=1$, $J=v_{22}=1$, $T=1+f(I,J)=1+f(1,2)=1+0=1$

$s=1$: $K=1$, $L=1$

$a=f(I,K)=f(1,1)=1$, $b=f(I,L)=f(1,1)=1$

$c=f(J,K)=f(2,1)=0$, $d=f(J,L)=f(2,1)=0$

$1-a=0$

$\alpha_{21}=Ta(1-bd)A(L,J)=(1-c)A(1,2)=A_{12}$

s=2: $K=1,\ L=2$

$a=f(I,K)=f(1,1)=1,\ b=f(I,L)=f(1,2)=0$

$c=f(J,K)=f(2,1)=0,\ d=f(J,L)=f(2,2)=1$

$\therefore\ 1-a=0$

$\therefore\ \alpha_{22}=T[dA(K,I)+a(1-bd)A(L,J)]=A(1,1)+A(2,2)=A_{11}+A_{22}$

s=3: $K=2,\ L=2$

$a=f(I,K)=f(1,2)=0,\ b=f(I,L)=f(1,2)=0$

$c=f(J,K)=f(2,2)=1,\ d=f(J,L)=f(2,2)=1$

$\therefore\ \alpha_{23}=TdA(K,I)=A(2,1)=A_{21}$

r=3: $I=v_{31}=2,\ J=v_{32}=2,\ \ T=1+f(I,J)=1+1=2$

s=1: $K=1,\ L=1$

$a=f(I,K)=0,\ b=f(I,L)=0$

$c=f(J,K)=0,\ d=f(J,K)=0$

$\therefore\ \alpha_{31}=0$

s=2: $K=1,\ L=2$

$a=f(I,K)=0,\ b=f(I,L)=0$

$c=f(J,K)=0,\ d=f(J,L)=1$

$\therefore\ 1-a=0$

$\therefore\ \alpha_{32}=TdA(K,I)=2A(1,2)=2A_{12}$

s=3: $K=2,\ L=2$

$a=f(I,K)=f(2,2)=1,\ b=f(I,L)=1$

$c=f(J,K)=1,\ d=f(J,L)=1$

$1-bd=1-a=0$

$\therefore\ \alpha_{33}=TdA(K,I)=2A(2,2)=2A_{22}$

Finally we have

$$\alpha \quad = \begin{bmatrix} 2A_{11} & 2A21 & 0 \\ A_{12} & A_{11}+A_{22} & A_{21} \\ 0 & 2A_{12} & 2A_{22} \end{bmatrix}$$

(II) Formula for finding matrix Ω. (APL function: PRP)

$$\Omega \;=\; [\;\omega_{rs}\;] \;, \qquad \begin{array}{l} r=1,2,\ldots,n(n+1)/2 \\ s=1,2,\ldots,n(n+1)(n^2+n+2)/8. \end{array}$$

The values of ω_{rs} is found by three steps:

(1) $I = v_{r1}$, $J=v_{r2}$ (the $r1^{th}$ and the $r2^{th}$ row of the matrix V)

(2) Construct a matrix U by writting the subscripts of the elements of vectors p. (APL function: INDEX1)

$$U = [\;u_{ij}\;] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 2 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & n \\ 1 & 1 & 2 & 2 \\ 1 & 1 & 2 & 3 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & n & n \\ 1 & 2 & 1 & 2 \\ \vdots & \vdots & \vdots & \vdots \\ n-1 & n & n-1 & n-1 \\ n-1 & n & n-1 & n \\ n & n & n & n \end{bmatrix} \qquad p = \begin{bmatrix} P_{11} & P_{11} \\ P_{11} & P_{12} \\ \vdots & \vdots \\ P_{11} & P_{1n} \\ P_{11} & P_{22} \\ P_{11} & P_{23} \\ \vdots & \vdots \\ P_{11} & P_{nn} \\ P_{12} & P_{12} \\ \vdots & \vdots \\ P_{n-1,n-1} & P_{n-1.n-1} \\ P_{n-1,n} & P_{n-1,n} \\ P_{n,n} & P_{n,n} \end{bmatrix}$$

where $i = 1,2,\ldots,n(n+1)(n^2+n+2)/8;$

$j = 1,2,3,\ldots$

Let $K=u_{s1}$, $L=u_{s2}$, $M=u_{s3}$, $N=u_{s4}$

(3) Define a function:

$$g(x,y,z) = \begin{cases} z & (x=y) \\ y & (x=z) \\ 0 & (x \neq y \text{ and } x \neq z) \end{cases}$$

for example,

$g(1,1,2)=2$, $g(1,1,1)=1$

$g(2,1,2)=1$, $g(1,2,2)=0$

Let $a=g(I,K,L)$, $b=g(I,M,N)$

$c=g(J,K,L)$, $d=g(J,M,N)$

and $T=1+f(K,M)f(L,N)$

Then ω_{rs} can be determined by

$$\omega_{rs} = [\ C(a,d)+C(b,c)\ ]/T \qquad (105)$$

when $l \times m=0$, $C(l,m)$ is defined as zero.

An illustrative example:

If $C = BR^{-1}B^T$ is a 2×2 matrix

$$C = \begin{bmatrix} C_{11} & C_{12} \\ C_{12} & C_{22} \end{bmatrix}$$

then Ω is a 3×6 matrix

$$\Omega = \begin{bmatrix} \omega_{11} & \omega_{12} & \omega_{13} & \omega_{14} & \omega_{15} & \omega_{16} \\ \omega_{21} & \omega_{22} & \omega_{23} & \omega_{24} & \omega_{25} & \omega_{26} \\ \omega_{31} & \omega_{32} & \omega_{33} & \omega_{34} & \omega_{35} & \omega_{36} \end{bmatrix}$$

(1)

$$V = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 2 & 2 \end{bmatrix}$$

(2)

$$U = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 2 \\ 1 & 1 & 2 & 2 \\ 1 & 2 & 1 & 2 \\ 1 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \end{bmatrix}$$

(3)

$r=1$: $I=v_{11}=1$, $J=v_{12}=1$

$s=1$: $K=u_{11}=1$, $L=u_{12}=1$, $M=u_{13}=1$, $N=u_{14}=1$

$a=g(I,K,L)=g(1,1,1)=1$, $b=g(I,M,N)=g(1,1,1)=1$

$c=g(J,K,L)=g(1,1,1)=1$, $d=g(J,M,N)=g(1,1,1)=1$

$T=1+f(K,M)f(L,N)=1+f(1,1)f(1,1)=1+1=2$

$\therefore \omega_{11}= [C(a,d)+C(b,c)]/T = C(1,1) = C_{11}$

$s=2$: $K=L=M=1$, $N=u_{24}=2$

$a=g(I,K,L)=g(1,1,1)=1$, $b=g(I,M,N)=g(1,1,2)=2$

$c=g(J,K,L)=g(1,1,1)=1$, $d=g(J,M,N)=g(1,1,2)=2$

$T=1+f(K,M)f(L,N)=1+f(1,1)f(1,2)=1+0=1$

$\therefore \omega_{12}= [C(a,d)+C(b,c)]/T = C(1,2)+C(2,1) = 2C_{12}$

$s=3$: $K=L=1$, $M=N=2$

$a=g(I,K,L)=g(1,1,1)=1$, $b=g(I,M,N)=g(1,2,2)=0$

$c=g(J,K,L)=g(1,1,1)=1$, $d=g(J,M,N)=g(1,2,2)=0$

$d=b=0$, $\therefore C(a,d)=C(b,c)=0$

$\therefore \omega_{13}=0$.

$s=4$: $K=u_{41}=1$, $L=u_{42}=2$, $M=u_{43}=1$, $N=u_{44}=2$

$a=g(I,K,L)=g(1,1,2)=2$, $b=g(I,M,N)=g(1,1,2)=2$

$c=g(J,K,L)=g(1,1,2)=2$, $d=g(J,M,N)=g(1,1,2)=2$

$T=1+f(K,M)f(L,N)=1+f(1,1)f(2,2)=1+1=2$

$\therefore \omega_{14}= [C(a,d)+C(b,c)]/T = C(2,2) = C_{22}$

$s=5$: $K=u_{51}=1$, $L=M=N=2$

$a=g(I,K,L)=g(1,1,2)=2$, $b=g(I,M,N)=g(1,2,2)=0$

$d=g(J,M,N)=g(1,2,2)=0$

$\therefore \omega_{15}= 0$

$s=6$: $K=L=M=N=2$

$a=g(I,K,L)=g(1,2,2)=0$, $b=g(I,M,N)=0$

$\therefore \omega_{16}= 0$

$r=2$: $I=v_{21}=1$, $J=v_{22}=2$

$s=1$: $K=L=M=N=1$

$c=g(J,K,L)=g(2,1,1)=0$, $d=g(J,M,N)=g(2,1,1)=0$

$\therefore\ \omega_{21}=0$

$s=2$: $K=L=M=1$, $N=2$

$a=g(I,K,L)=g(1,1,1)=1$, $b=g(I,M,N)=g(1,1,2)=2$

$c=g(J,K,L)=g(2,1,1)=0$, $d=g(J,M,N)=g(2,1,2)=1$

$T=1+f(K,M)f(L,N)=1+f(1,1)f(1,2)=1+0=1$

$\therefore\ \omega_{22}=C_{11}$

$s=3$: $K=L=1$, $M=N=2$

$a=g(I,K,L)=g(1,1,1)=1$, $b=g(I,M,N)=g(1,2,2)=0$

$d=g(J,M,N)=g(2,2,2)=2$

$T=1+f(K,M)f(L,N)=1+f(1,2)f(1,2)=1$

$\therefore\ \omega_{23}=C(a,d)=C(1,2)=C_{12}$

$s=4$: $K=1$, $L=2$, $M=1$, $N=2$

$a=g(I,K,L)=g(1,1,2)=2$, $b=g(I,M,N)=g(1,1,2)=2$

$c=g(J,K,L)=g(2,1,2)=1$, $d=g(J,M,N)=g(2,1,2)=1$

$T=1+f(K,M)f(L,N)=1+f(1,1)f(2,2)=1+1=2$

$\therefore\ \omega_{24}=[C(a,d)+C(b,c)]/T = C(2,1) = C_{21}$

$s=5$: $K=1$, $L=M=N=2$

$a=g(I,K,L)=g(1,1,2)=2$, $b=g(I,M,N)=g(1,2,2)=0$

$d=g(J,M,N)=g(2,2,2)=2$

$T=1+f(K,M)f(L,N)=1+f(1,2)f(2,2)=1$

$\therefore\ \omega_{25}=C(a,d)=C(2,2)=C_{22}$

$s=6$: $K=L=M=N=2$

$a=g(I,K,L)=g(1,2,2)=0, \quad b=g(I,M,N)=0$

$\therefore \omega_{26}= 0$

$r=3$: $\quad I=v_{31}=2, \quad J=v_{32}=2$

$s=1$: $K=L=M=N=1$

$a=g(I,K,L)=g(2,1,1)=0,$

$c=g(J,K,L)=g(1,2,2)=0,$

$\therefore \omega_{31}= 0$

$s=2$: $K=L=M=1, \quad N=2$

$a=0,$

$c=0$

$\therefore \omega_{32}= 0$

$s=3$: $K=L=1, \quad M=N=2$

$a=0,$

$c=0$

$\therefore \omega_{33}=0.$

$s=4$: $K=1, \quad L=2, \quad M=1, \quad N=2$

$a=g(I,K,L)=g(2,1,2)=1, \quad b=g(I,M,N)=g(2,1,2)=1$

$c=g(J,K,L)=g(2,1,2)=1, \quad d=g(J,M,N)=g(2,1,2)=1$

$T=1+f(K,M)f(L,N)=1+f(1,1)f(2,2)=1+1=2$

$\therefore \omega_{34}= [C(a,d)+C(b,c)]/T = C(1,1) = C_{11}$

s=5: K=1, L=M=N=2

a=g(I,K,L)=g(2,1,2)=1, b=g(I,M,N)=g(2,2,2)=2

d=g(J,M,N)=g(2,1,2)=1, d=g(J,M,N)=g(2,2,2)=2

T=1+f(K,M)f(L,N)=1+f(1,2)f(2,2)=1

$\therefore\ \omega_{35} = C(2,1)+C(1,2) = 2C_{12}$

s=6: K=L=M=N=2

a=b=c=d=2

T=1+f(K,M)f(L,N)=1+f(2,1)f(2,2)=1+1=2

$\therefore\ \omega_{36} = C_{22}$

we finally obtain

$$\Omega = \begin{bmatrix} C_{11} & 2C_{12} & 0 & C_{22} & 0 & 0 \\ 0 & C_{11} & C_{12} & C_{21} & C_{22} & 0 \\ 0 & 0 & 0 & C_{11} & 2C_{12} & C_{22} \end{bmatrix}$$

(III) Function package for solving equation (99)

(1) APL function named RIC,

(2) APL function named AFA,

(3) APL function named INDEX,

(4) APL function named IFF,

(5) APL function named PRP,

(6) APL function named INDEX1,

(7) APL function named FUN,

(8) APL function named WITH,

(9) APL function named UPT,

(10) APL function named PDOT,

(11) APL function named FORMPP,

(12) APL function named FORM.

the p matrix is found by calling the APL function named RIC.

```
    ∇ P←RIC;PV;W;Q1;Q2;Q3;Q4;I;G
   ⍝ SOLUTION TO RICCATI EQUATIONS
    'ENTER N---- DIMENSION OF SYSTEM MATRIX A :'
    N←⎕
    'ENTER MATRIX A :'
    F←AFA A←(N,N)⍴⎕
    'ENTER M---- DIMENSION OF MATRIX R :'
    M←⎕
    'ENTER MATRIX R :'
    R←(M,M)⍴⎕
    'ENTER INPUT MATRIX B :'
    W←PRP G←B+.×(⌹R)+.×⍉B←(N,M)⍴⎕
    'ENTER VECTOR QV=Q11,Q12,...,Q1N,Q22,...Q2N,Q33,.....,QNN :'
    QV←⎕
    'ENTER STARTING TIME TN :'
    TN←⎕
    'ENTER INITIAL VALUES OF VECTOR P :'
    PV←UPT(N,N)⍴⎕
    'ENTER TIME INCREMENT ΔT ( NEGATIVE VALUES SHOULD BE USED ):'
    ΔT←⎕
    'ENTER THE NUM. OF POINTS OF SOLUTIONS OF RUNGE-KUTTA METHOD:'
    NN←⎕
    I←1
   L:T←TN+ΔT
    Q4←ΔT×PDOT PV+Q3←ΔT×PDOT PV+0.5×Q2←ΔT×PDOT PV+0.5×Q1←ΔT×PDOT PV
    PV←PV+(÷6)×Q1+Q4+2×Q2+Q3
    →L IFF NN≥I←I+1
    'THE RESULT---- MATRIX P IS:'
    ⎕←P←N FORMP PV
    ∇


    ∇ F←AFA A;R;S;N;M;I;J;K;L;V;B;C;D;E;H;T
   ⍝ FORMING MATRIX F=[α]
    V←INDEX N←(R←1)↑⍴A
    F←(M,M←0.5×N×1+N)⍴0
   L0:I←V[R;S←1]
    T←1+I=J←V[R;2]
   L1:C←J=K←V[S;1]
    H←(1-C)×B←I=L←V[S;2]
    E←1-B×D←J=L
    F[R;S]←T×(A[K;I]×D)+(A[L;J]×(I=K)×E)+((A[L;I]×C)+A[K;J]×H)×(I≠K)×~
    D
    →L1 IFF M≥S←S+1
    →L0 IFF M≥R←R+1
    ∇


    ∇ V←INDEX N;I;R
    V←(2,N)⍴(N⍴I←1),R←⍳N
   L:V←V,[2](2,N-I)⍴((N-I)⍴I+1),I↓R
    →L IFF(N-1)≥I←I+1
    V←⍉V
    ∇


    ∇ A←B IFF C
    A←C⍴B
    ∇
```

```
    ∇ W←PRP G;N;M;I;J;K;L;R;S;A;B;C;D;V;U;V1;U1;MM;NN;T
   ⍝ G=B+.×(⌹R)+.×⍉B, G IS A SYMMETRICAL MATRIX
    MM←1↑ρV←INDEX N←(R←1)↑ρG
    NN←1↑ρU←INDEX1 N
    W←(MM,NN)ρ0
   L0:J←V1[2] WITH I←(S←1)↑V1←V[R;]
   L1:K←1↑U1←U[S;]
    A←I FUN K,L←U1[2]
    C←J FUN K,L
    M←U1[3]
    T←1+(K=M)∧L=N←U1[4]
    B←I FUN M,N
    →BR1 IFF 0≠A×B×C×D←J FUN M,N
    →BR0 IFF(0=A×D)∧0=B×C
    →BR2 IFF 0=A×D
    →BR0 WITH W[R;S]←G[A;D]÷T
   BR2:→BR0 WITH W[R;S]←G[B;C]÷T
   BR1:W[R;S]←(G[A;D]+G[B;C])÷T
   BR0:→L1 IFF NN≥S←S+1
    →L0 IFF MM≥R←R+1
    ∇

    ∇ U←INDEX1 N;I;V;M
    M←1↑ρV←INDEX N
    U←((M,I←2)ρ1),[2] V
   L:U←U,[1]((I-1),0)↓((M,2)ρV[I;]),[2] V
    →L IFF M≥I←I+1
    ∇

    ∇ F←X FUN YANDZ;Y;Z
    Y←1↑YANDZ
    F←((Z×X=Y)+Y×X=Z)-Y×(X=Y)∧X=Z←¯1↑YANDZ
    ∇

    ∇ A←B WITH C
    A←B
    ∇
    ∇UPT[⎕]∇
    ∇ VA←UPT A;I;N
    VA←⍳0×N←(I←1)↑ρA
   L:VA←VA,A[I;(I-1)+⍳N+1-I]
    →L IFF N≥I←I+1
    ∇

    ∇ PD←PDOT PV
    PD←(W+.×FORMPP PV)-(F+.×PV)+QV
    ∇

    ∇ PP←FORMPP PV;M1;M2;Q
    M2←((⍳Q)∘.≤⍳Q)×M1←(Q,Q←ρPV)ρPV
    PP←UPT M2×⍉M1
    ∇

    ∇ P←N FORMP PV;B;I
   ⍝ FORMING SYMMETRICAL MATRIX P FROM VECTOR PV
    P←(N,N)ρ¯1+I←1
   L:P[I;(I-1)+⍳B]←(B←N+1-I)↑PV
    PV←B↓PV
    →L IFF N≥I←I+1
    P←P+(⍉P)×0≠(⍳N)∘.-⍳N
    ∇
```

(IV) Illustrative example.

For the given plant

$$x = \begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

and for the quadratic performance index

$$J = \frac{1}{2} \int_0^{\infty} \{ x^T \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} x + u^T (1/2) u \} dt$$

the input data are

N=2, M=1, TN=0, NN=100, ΔT=-0.05

$$A = \begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad R=1/2,$$

$$Q = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

The initial values of p = [0,0,0].

```
      P4RIC
ENTER N---- DIMENSION OF SYSTEM MATRIX A :
⎕:
      2
ENTER MATRIX A :
⎕:
      0 1 0 ¯1
ENTER M---- DIMENSION OF MATRIX R :
⎕:
      1
ENTER MATRIX R :
⎕:
      0.5
ENTER INPUT MATRIX B :
⎕:
      0 1
ENTER VECTOR QV=Q11,Q12,...,Q1N,Q22,...Q2N,Q33,.....,QNN :
⎕:
      2 0 2
ENTER STARTING TIME TN :
⎕:
      0
ENTER INITIAL VALUES OF VECTOR P :
⎕:
      0 0 0
ENTER TIME INCREMENT ΔT ( NEGATIVE VALUES SHOULD BE USED ):
⎕:
      ¯0.05
ENTER THE NUM. OF POINTS OF SOLUTIONS OF RUNGE-KUTTA METHOD:
⎕:
      100
THE RESULT---- MATRIX P IS:
 3 1
 1 1
```

7.0 Conclusions

We have proposed an effective way to compute symbolic system functions using the Grassmann algebra in the single input case and extend the technique to the multi-input case. The main idea is to take the advantage of the sparse nature of a dynamic system and the elegant notations of Grassmann algebra.

In the graph representation, we followed Coates' sense, that is the outgoing and incoming branches are corresponding the column vector and row vector of a matrix respectively. However, the method is much simpler than coates' approach.

For a hybrid system which contains analog as well as digital notes, a new formula has been established. This technique is particularly useful in frequently encountered computer-controlled systems.

A fundamental technique in multivariable systems analysis and design is decoupling. The P-constrained and V-constrained decoupling structures are studied in detail under the light shed by the Grassmann algebra.

The typical and well known Smith pre-estimator and controller is investigated, particularly in the multivariable case by use of idea of Grassmann.

Finally, for showing the power of Grassmann, we applied to the solutions of two basic equations. One is the Liapunov matrix equation for stability studies and one is the Riccati matrix equations for optimal control.

8.0 Appendices

These appendices present two other methods for symbolic evaluations. By comparing them with the Grassmann Algebra method, we can conclude that the Grassmann method is more elegant and easier to use.

A1.0 Another Method for Symbolic Evaluation.

To accomplish symbolic evaluations, we can use methods other than Grassmann algebra. This chapter is to present a symbolic evaluation method based on the Fourier transforms. By comparing the Fourier transform methodology with that of the Grassmann approach, it finds that the later one is more simple in manipulations and applicable to evaluation other than transfer functions.

A1.1 Symbolic Evaluation Via Fourier Transform.

Lee[16] developed a Fourier method to determine the characteristic function

$$f(s) = \det[\ sI-A\ ] = s^N + \alpha_{N-1}s^{N-1} + \ldots + \alpha_1 s + \alpha_0 \qquad (106)$$

from a state space model

$$x = Ax \qquad (107)$$

The method can be summarized as follows:

Let $W = \exp\ (2\pi/N)j$ be the N^{th} root of unity, then from (106) we obtain

$$f(W^{-m}) = \det[\ W^{-m}I-A\ ] \qquad (108)$$

$$= W^{-mN} + \alpha_{N-1}W^{-m(N-1)} + \ldots + \alpha_1 W^{-m} + \alpha_0 \qquad (109)$$

$$m = 0,1,\ldots,N-1$$

Since m and N are integers, therefore

$$W^{-mN} = (W^N)^{-m} = \exp\,(-2m\pi)j = 1$$

From (109), we have

$$\det[\ W^{-m}I - A\ ] - 1 = \alpha_{N-1}W^{-m(N-1)} + \ldots + \alpha_1 W^{-m} + \alpha_0 \quad (110)$$

Let

$$a_m = \det[\ W^{-m}I - A\ ] - 1 \quad (111)$$

from (110) and (111), we have

$$a_m = \sum_{k=0}^{N-1} \alpha_k W^{-mk} \qquad \text{for } m = 0,1,2,\ldots,N-1 \quad (112)$$

Therefore

$$\alpha_k \overset{N}{\longleftrightarrow} a_m$$

forms a discrete Fourier series pair

The inversion formula gives that

$$\alpha_k = \frac{1}{N} \sum_{m=0}^{N-1} a_m W^{mk} \qquad \text{for } k=0,1,2,\ldots,N-1 \quad (113)$$

Now we can extend Lee's Fourier method to transfer function evaluation by use of (14)

Illustrative example:

For the given system

$$\dot{x} = \begin{bmatrix} -1 & -2 & -2 \\ 0 & -1 & 1 \\ 1 & 0 & -1 \end{bmatrix} x + \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix} u$$

$$y = [\ 1,\ 1,\ 0\ ]\ x$$

its corresponding transfer function is desired.

We first evaluate the characteristic equation

$$f(s) = \det[\ sI-A\ ]$$

$$= \begin{bmatrix} s+1 & 2 & 2 \\ 0 & s+1 & -1 \\ -1 & 0 & s+1 \end{bmatrix}$$

Let N=3, then $W = e^{\,j(2\pi/3)}$

substituting into (111) gives

$$a_m = \det[\ W^{-m}I-A\ ] - 1$$

$$= \begin{bmatrix} e^{-j(2m\pi/3)} + 1 & 2 & 2 \\ 0 & e^{-j(2m\pi/3)} + 1 & -1 \\ -1 & 0 & e^{-j(2m\pi/3)} + 1 \end{bmatrix} - 1$$

Then we have

$$a_0 = 13$$

$$a_1 = 1 - j1.732$$

and $$a_2 = 1 + j1.732$$

using the inverse formula (113), we have

$$\alpha_k = \frac{1}{N} \sum_{m=0}^{N-1} a_m W^{mk} = \frac{1}{3} \sum_{m=0}^{2} a_m \, e^{j(2mk\pi/3)} \quad (k=0,1,2)$$

therefore

$$\alpha_0 = 5$$

$$\alpha_1 = 5$$

$$\alpha_2 = 3$$

The characteristic equation is obtained as follows

$$f(s) = S^3 + 3s^2 + 5s + 5 \tag{114}$$

Then we try to find the function:

$$f(s) = \det[\ (sI-A) + bc^T\ ]$$

$$= \begin{bmatrix} s+3 & 4 & 2 \\ 0 & s+1 & -1 \\ 0 & 1 & s+1 \end{bmatrix}$$

It is still a determinant evaluation problem.

Again let N=3 and $W = e^{\, j(2\pi/3)}$

substituting into (111) gives

$$a_m = \det[\, W^{-m}I - A \,] - 1$$

$$= \begin{bmatrix} e^{-j(2m\pi/3)} + 3 & 4 & 2 \\ 0 & e^{-j(2m\pi/3)} + 1 & -1 \\ 0 & 1 & e^{-j(2m\pi/3)} + 1 \end{bmatrix} - 1$$

After evaluation:

$$a_0 = 19$$

$$a_1 = -0.5 - j2.598$$

and $a_2 = -0.5 + j2.598$

using the inverse formula (113), we have

$$\alpha_k = \frac{1}{N} \sum_{m=0}^{N-1} a_m W^{mk} = \frac{1}{3} \sum_{m=0}^{2} a_m e^{j(2mk\pi/3)} \quad (k=0,1,2)$$

therefore

$$\alpha_0 = 6$$

$$\alpha_1 = 8$$

$$\alpha_2 = 5$$

Therefore $\det[\,(sI-A) + bc^T\,] = s^3 + 5s^2 + 8s + 6$ (115)

substituting (114) and (118) into (14) gives

$$c^T(sI-A)^{-1}b = \frac{s^3 + 5s^2 + 8s + 6}{s^3 + 3s^2 + 5s + 5} - 1$$

$$= \frac{2s^2 + 3s + 1}{s^3 + 3s^2 + 5s + 5}$$

which is our desired transfer function.

A program based on the discrete Fourier method to evaluate the corresponding transfer function of a state space model is written. The language used is APL; the program involves symbolic determinant evaluation only. All the non-zero elements in the two determinants are complex in general. The program and outputs for the illustrative example are given in the next pages

```
      ∇TRANSFER[⎕]∇
      ∇ TRANSFER
[1]   ⍝ EVALUATING TRANSFER FUNCTIONS FROM STATE SPACE MODELS
[2]   ⍝ VIA FOURIER TRANSFORM
[3]   ⍝ FROM DX=AX+BU AND Y=CTX TO Y(S)/U(S) DETERMINATION
[4]    'STEP 1. EVALUATE THE COEFS. OF DET(SI-A)'
[5]    ' '
[6]    DC←CHARA A
[7]    ' '
[8]    'STEP 2. EVALUATE THE COEFS. OF  DET(SI-A+BCT)'
[9]    ' '
[10]   N1←CHARA A-B+.×⍉C
[11]   ' '
[12]   'THE COEFS. OF NUMERATOR  ARE :          ',⍕N1-DC
[13]   'THE COEFS. OF DENOMINATOR ARE:          ',⍕DC
      ∇
      ∇CHARA[⎕]∇
      ∇ CF←CHARA A;N;I;X;X1;S0;SI;M;V;BM;AK;U;J;CA
[1]    X←-(○2)÷N←1↑⍴A
[2]    BM←AK←(M←2×N)⍴I←0
[3]    CA←(V←M⍴ 1 0)\A
[4]   L:U←(2×I)+⍳2×I+1
[5]    S0←(2○X1),1○X1←I×X
[6]    SI←(N,M)⍴S0,M⍴0
[7]    BM[U]←(¯1 0)+CDET SI-CA
[8]    →((N-1)≥I←I+1)/L
[9]    'A(0),A(1),....,A(N-1) : ( COMPLEX NUMBERS )'
[10]   'REAL PART  IMAG. PART'
[11]   ⎕←(N,2)⍴BM
[12]   I←0
[13]  L1:J←0
[14]   U←(2×I)+⍳2×I+1
[15]  L2:AK[U]←AK[U]+BM[(2×J)+⍳2×J+1] CMP S0←(2○X1),1○X1←-X×I×J
[16]   →((N-1)≥J←J+1)/L2
[17]   →((N-1)≥I←I+1)/L1
[18]   ' '
[19]   'THE COEFS. ARE IN DESCENDING ORDER :',⍕CF←⌽(V/AK÷N),1
      ∇
      ∇CDET[⎕]∇
      ∇ Z←CDET A;B;P;I;U;N;K;A;V
[1]    U←(¯1 0)+2×I←1
[2]    Z← 1 0
[3]   L:AI←'MODULUS ' CREPEAT,A[;U]
[4]    →(I=P←AI⍳⌈/AI)/LL
[5]    A[I,P;]←A[P,I;]
[6]    Z←-Z
[7]   LL:Z←Z CMP B←A[I;U]
[8]    →((1=×/Z=0)∨1=N←1↑V←⍴A)/0
[9]    →(1=×/B= 0 0)/BR1
[10]   B←(1 ¯1)×B÷+/B*2
[11]  BR1:AA←'B CMP' CREPEAT,A[;U]
[12]   A←V⍴¯1+K←1
[13]  LB:A[K;]←'(2↑(2×K-1)↓AA) CMP ' CREPEAT A[I;]
[14]   →(N≥K←K+1)/LB
[15]   A← 1 2 ↓A-A
[16]   →L
[17]  ⍝EVALUATES A COMPLEX DETERMINANT
      ∇
```

```
      ∇MODULUS[⎕]∇
      ∇ R←MODULUS Z
[1]     R←(+/Z*2)*0.5
      ∇
      ∇CREPEAT[⎕]∇
      ∇ CV←FUNCNAME CREPEAT ARGVEC
[1]     CV←⍳0
[2]     →(0=⍴ARGVEC)/0
[3]     CV←⍎FUNCNAME,' 2↑ARGVEC'
[4]     CV←CV,FUNCNAME CREPEAT 2↓ARGVEC
      ∇
      ∇CMP[⎕]∇
      ∇ Z←P CMP Q
[1]     Z←(-/P×Q),+/P×⌽Q
[2]   ⍝P AND Q ARE COMPLEX NUMBERS
      ∇
```

Illustrative Example

```
      A←3 3⍴¯1 ¯2 ¯2 0 ¯1 1 1 0 ¯1

      B←3 1⍴2 0 1

      C←3 1⍴1 1 0

      TRANSFER

STEP 1. EVALUATE THE COEFS. OF DET(SI-A)

A(0),A(1),....,A(N-1) : ( COMPLEX NUMBERS )
REAL PART  IMAG. PART
  13        0
   1       ¯1.732
   1        1.732

THE COEFS. ARE IN DESCENDING ORDER :1 3 5 5

STEP 2. EVALUATE THE COEFS. OF  DET(SI-A+BCT)

A(0),A(1),....,A(N-1) : ( COMPLEX NUMBERS )
REAL PART  IMAG. PART
  17       0
 ¯0.5    ¯2.598
 ¯0.5     2.598

THE COEFS. ARE IN DESCENDING ORDER :1 5 8 6

THE COEFS. OF NUMERATOR  ARE :        0 2 3 1
THE COEFS. OF DENOMINATOR ARE:        1 3 5 5
```

A2.0 Symbolic Function Generation Using Number Theoretic Transform.

A2.1 Introduction.

This appendix shows that orthogonal functions which can be generated by a cyclic group under multiplication are suitable for the generation of symbolic network functions in computer-aided analysis and design of electronic circuits. A multi-dimensional Number Theoretic Transform is defined which gives an efficient method to generate symbolic network functions without quantization error.

a network function is usually obtained from a physical model of the electronic circuit concerned. In general, the function may be represented by a ratio of polynomials in L, C, the complex frequency s and other parameters. These polynomials are the expansions of two determinants which are found by nodal, loop or other kinds of network analysis. This shows that the problem of generating symbolic network functions of a given network reduces to the problem of generating the expanded result of given determinants.

$$Y(x_1,x_2,\ldots,x_q) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} \cdots \sum_{n_q=0}^{N_q-1} c_{n1,n2,\ldots,nq} x_1^{n1} x_2^{n2} \ldots x_q^{nq} \tag{116}$$

where $x_1,x_2,\ldots,x_q$ are symbolic entries of L,C,s,...

and $N_1-1, N_2-1, \ldots N_q-1$ are maximum possible $n_1,n_2,\ldots,n_q$ respectively.

A2.2 Orthogonal Functions.

We assume that N functions $\phi_{k,0}(m_k)$, $\phi_{k,1}(m_k)$, ..., $\phi_{k,N-1}(m_k)$ are mutually orthogonal between 0 and (N-1). The function

$$Y(m_1,m_2,\ldots,m_q) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} \cdots \sum_{n_q=0}^{N_q-1} c_{n1,n2,\ldots,nq}\phi_{1,n1}(m_1)\phi_{2,n2}(m_2)\cdots\phi_{q,nq}(m_q) \tag{117}$$

can be represented by choosing the c's according to the following equation:

$$c_{n1,n2,\ldots,nq} = \sum_{m_1=0}^{N_1-1} \sum_{m_2=0}^{N_2-1} \cdots \sum_{m_q=0[}^{N_q-1} Y(m_1,m_2,\ldots,m_q)\phi_{1,n1}^{-1}\ \phi_{2,n2}^{-1} \cdots \phi_{q,nq}^{-1} \tag{118}$$

Equation (116) is a general representation of the symbolic network function of a given network. In order to make equation (117) to be compatible with equation (116), let x_k^{nk} be represented by $\phi_{k,nk}$. Since the set $\{x_k^{nk}:n_k=0,1,\ldots N-1\}$ can be generated by x_k^0, $x_k^0.x_k$, $x_k.x_k$, $x_k^2.x_k$, $x_k^3.x_k$, ... The orthogonal function $\phi_{k.nk}$ sets must also be generated by a cyclic group under multiplication, say α_k's for example. That is

$$(\alpha_k)^n = \phi_{k,nk}$$

This forms the basic criterion used in this paper for employing orthogonal functions to find symbolic network functions to find symbolic network functions. The coefficients $c_{n1,n2,\ldots,nq}$ in equation (116) is equivalent to the corresponding coefficients in equation (117) if x_k^{nk} is replaced by $\phi_{k,nk}(m_k)$. All combinations of $Y(m_1,m_2,\ldots,m_q)$ could be calculated by

substituting ${x_n}^{kn}$ by $\phi_{k,nk}(m_k)$ with all possible combinations of $m_1, m_2, \ldots, m_q$ in the determinant under question. Hence the coefficients $c_{n1,n2,\ldots nq}$'s could be found from equation (118).

For the complex number field, $\alpha_n = e^{-j2\pi/N}$ is the generator for a set of orthogonal function used to generate symbolic network functions. This formulation is equivalent to doing the calculation by multi-dimensional Discrete Fourier transforms(DFT). In a finite ring, we may also find a cyclic group under multiplication modulo a prime number to form an Number Theoretic Transform which may be used to generate the network function. If we use the DFT to generate the symbolic network functions, calculation error can hardly be avoided, since both the DFT and the Inverse DFT involve complicated multiplications of irrational, complex numbers. In next section, we discuss to use the Number Theoretic Transform to generate symbolic functions. This new method gives an error-free calculation of symbolic functions.

A2.3 Generation Via Number Theoretic Transform(NTT).

Let M be the modulo base of the NTT to be used. A multi-dimensional NTT pair can now be defined as

$$Y(m_1, m_2, \ldots, m_q) = \left\langle \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} \ldots \sum_{n_q=0}^{N_q-1} a(n_1, n_2, \ldots n_q) \alpha^{m1n1} \alpha^{m2n1} \ldots \alpha^{mqnq} \right\rangle_M \tag{119}$$

and

$$a(n_1,n_2,\ldots n_q)=\langle N_1^{-1}N_2^{-1}\ldots N_q^{-1}\sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1}\ldots\sum_{n_q=0}^{N_q-1}$$

$$\alpha^{-m1n1}\alpha^{-m2n2}\ldots\alpha^{-mqnq}\rangle_M \tag{120}$$

where $n_1,n_2,\ldots,n_q=0,1,\ldots,(N_1-1),\ldots,(N_q-1)$ respectively.

To demonstration the application of the NTT to symbolic network function generation, let us find the input impedance of the circuit shown in Figure 41. The mesh matrix of the circuit can be written as

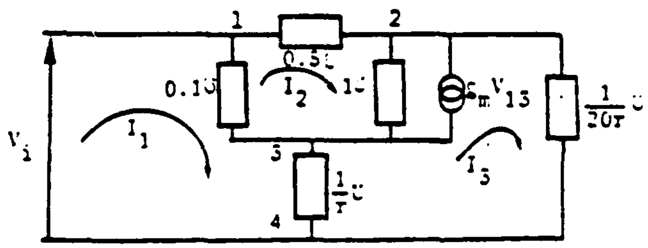


Figure 41

$$\begin{bmatrix} V_i \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 10+r & -10 & -r \\ -10(1+g_m) & 13+10g_m & -1 \\ 10g_m-r & -(1+10g_m) & 1+21r \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} \tag{121}$$

where r and g_m are network symbols.

Let M=65537, $N_1=N_2=4$ and the root of unity, α, be 2^8. Equation (119) and (120) become

$$Y(m_1,m_2)=\left\langle \sum_{n_1=0}^{3} \sum_{n_2=0}^{3} a(n_1,n_2)2^{8(m1n1+m2n2)}\right\rangle_{65537} \qquad (122)$$

and

$$a(n_1,n_2)=\left\langle 4^{-1}4^{-1} \sum_{m_1=0}^{3} \sum_{m_2=0}^{3} Y(m_1,m_2)2^{-8(m1n1+m2n2)}\right\rangle_{65537} \qquad (123)$$

substitute r by 2^{8m1} and g_m by 2^{8m2} into equation (121),

$$\Delta(r,g_m) \implies \Delta(2^{8m1},2^{8m2}) =$$

$$\left\langle \begin{vmatrix} 10+2^{8m1} & -10 & -2^{8m1} \\ -10(1+2^{8m2}) & 13+10.2^{8m2} & -1 \\ 10.2^{8m2}-2^{8m1} & -(1+10.2^{8m2}) & 1+21.2^{8m1} \end{vmatrix} \right\rangle_{65537} \qquad (124)$$

for m_1,m_2=0,1,2,3 respectively.

Equation (124) is actually a physical representation of the general equation (119) or equation (122), hence

$$\Delta(2^{8m1},2^{8m2}) = Y(m_1,m_2) \text{ and}$$

$\{Y(m_1,m_2):m_1,m_2=0,1,2,3\}$ = { 1122,57222,682,10119,32838,42235,22998,13601,
65375,45738,65015,19115,31819,51496,42459,22782 }

The inverse transform can be found by equation (123) and this gives the coefficients of the expansion of equation (121).

$\{a(n_1,n_2):n_1,n_2=0,1,2,3\}$ = { 20,0,0,0,622,20,0,0,260,200,0,0,0,0,0,0}

Hence $\Delta(r,g_m) = 20+622r+260r^2+20rg_m+200r^2g_m$

Similarly, Δ_{11} can be found by the same procedure. We obtain

$$Z_{in} = \frac{V_i}{I_1} = \frac{\Delta}{\Delta_{11}} = \frac{20+622r+260r^2+20rg_m+200r^2g_m}{12+273r+210rg_m}$$

A2.4 Conclusion.

The number Theoretic Transform gives a completely error free calculation of symbolic network functions. The major difficulty of the NTT is that the dynamic range is limited by the base of the module arithmetic involved. This problem can be overcome by using a large modulus as the base of using the Chinese Remainder Theorem.

9.0 References

[1] Karpopp, D.,"Lagrange's Equations for Complex Bond Graph System", ASME Journal of Dynamic System, Measurement and Control, Vol.99, No. 4, Dec. 1977, pp 300-306 and "The Energetic Structure of Multibody Systems", Joural of Franklin Institute, Vol. 306, No. 2, August 1978, pp. 317-342.

[2] Chen, C.F. and I.J. Haas,"Elements of Control Systems: Classical and Modern Approaches", Prentice Hall, 1968.

[3] Gibson, J.E.,"Nonlinear Control System", McGraw Hill Co., 1963

[4] Ku, Y.H. and C.F. Chen,"Stability Study of a Third Order Servomechanism with Multiplicative Feedback Control", American Institute for Electrical Engineers Transactions on Applications and Industry, July, 1958.

[5] Chua, L.O. and P.M. Lin,"Computer-aided Analysis of Electronic Circuits, Algorithms and Computational Techniques", Prentice Hall, 1975.

[6] Alderson, G.E. and P.M. Lin,"Computer Generation of Symbolic Network functions", IEEE Transaction on AC-20, 1973, pp48-56, and Singhal, K. and J. Vlach,"symbolic Circuit Analysis", IEEE, April, 1981.

[7] Lin, P.M.,"A Survey of Applications of symbolic Network functions", IEEE Transactions on Circuit Theory, Vol. CT-20, pp. 732-737, November 1973.

[8] Shien, S.D. and S.P. Chan,"Topological Formulation of symbolic Network

Functions and Sensitive Analysis of Active Networks", IEEE Transactions on Circuit Theory, Vol. CAS-21, pp.39-45, January 1974.

[9] Lin, P.M. and G.E. Alderson,"Computer Generation of Symbolic Network Functions--A New Theory and Implementation", IEEE Transactions on Circuit Theory, Vol.CT-20, pp. 48-56, January 1973.

[10] Whitehead, A.N.,"A Treatise on Universal Algebra with Applications", Cambridge University Press, 1977, reprinted by Hafner Publishing Co.

[11] Sedlar, M. and G.A. Bekey.,"Signal Flow Graphs of Sampled-data Systems", IEEE AC-12, No. 2, April 1967, 154-161.

[12] Kuo, B.C.,"Digital Control Systems", Holt, Rinehart and Winston Inc., 1980.

[13] Mesarovic, M.D.,"The Control of Multivariable Systems", John Wiley, 1960.

[14] Liu, C.H.,"General Decoupling Theory of Multivariable Process Control Systems", Hydroelectric Press, 1984.

[15] Fan, X.,"The Output Equation of discrete-Continuous Hybrid systems--An Extension of Mason's Formula", Acta Automatica sinica, Vol. 11, No. 4, October 1985, pp. 433-437.

[16] Lee, T.,"A simple Method to Determine the Characteristic Function f(s) = |sI-A| by discrete Fourier Series & Fast Fourier Transform", IEEE Transactions on Circuits and Systems, April 1976, pp.242.

[17] Chen, C.F., Y.T. Tsay and R.E. Yates,"A Unified Approach to Deadbeat Systems Design", Computer & Electrical Engineering, Vol. 7, pp.111-129, 1980.

[18] Gantmacher, F.R.,"The Theory of Matrices", Vol. 1, Chelsea Publishing Co., N.Y., 1960.

[19] Yeung, K.S.,"Symbolic Network Function generation via Discrete Fourier Transform",IEEE Tran., Vol. CAS-31, pp. 229-231,1984.

[20] McClellan, J.H. and C.M. Rader.,"Number Theory in Digital Signal Processing", prentice Hall Inc., 1979.

[21] Schizas, C. and F.J Evans.,"APL And Graph Theory in Dynamic systems Analysis", IEE Proc., Vol.128, Pt. D, No. 3, May 1981, pp. 85-92.

[22] Evans, F.J., Schizas, C., and Chan, J.,"Control System Design Using Graphical Decomposition Techniques", ibid, pp. 77-84.

[23] Siu, W.C. and C.F. Chen,"A New Technique For Symbolic Function Generation Using Number Theoretic Transform", IEEE Int. Symposium on Circuits and System, Kyoto, Japan, June 1985.

[24] Chen, C.F., and M.B. Ahmad,"Evaluating The Gain of a Flow Graph by the Grassmann Algebra", Int. J. Control, 1984, Vol.39, No.6, pp.1329-1337.

[25] Sannuti, P., and N.N. Puri,"Symbolic Network Analysis -- An Algebraic Formulation", IEEE Trans. Circuit syst., Vol. 27, 1980, pp. 679-.

[26] Mielke, R.R.,"A New Signal Flowgraph Formulation of Symbolic Network Function", IEEE Trans., Vol. CAS-25, No.6, June 1978, pp.334-340.

[27] Lee, B.G.,"The Product Matrices and New Gain Formulas", IEEE Trans., Vol. CAS-27, No.4, April 1980, pp.284-292.

[28] Riegle, D.E.,and P.M. Lin,"Matrix Signal Flow Graphs and an Optimum topological Method for Evaluation their Gains", IEEE Trans. Vol. CT-19, No.5, Sept. 1972, pp.427-435.

DISTRIBUTION

	No. of Copies
US Army Materiel System Analysis Activity ATTN: AMXSY-MP Aberdeen Proving Cround, MD 21005	1
IIT Research Institute ATTN: GACIAC 10 W. 35th Street Chicago, IL 60616	1
AMSMI-RD, Dr. McCorkle	1
Dr. Rhoades	1
AMSMI-RD-GC, Mr. Bradford	2
AMSMI-RD-CS-R	15
AMSMI-RD-CS-T	1
AMSMI-GC-IP, Mr. Bush	1